VNIVERSITAT DE VALÈNCIA

Departamento de Física Atómica, Molecular y Nuclear
IFIC (Universitat de València - CSIC)

The Back-End Electronics for the ATLAS Hadronic Tile Calorimeter at the Large Hadron Collider

TESIS DOCTORAL

Jose Alberto Valero Biot

DIRECTORES
Juan A. Valls Ferrer
Vicente González Millán

Valencia, 2014

Declaration

This dissertation is the result of my own work, except where explicit reference to the work of others is made. It has not been submitted for another qualification in this or any other university.

Jose Alberto Valero Biot

Preface

The work reported in this thesis was carried out during the first three years of operation (2010-2012) of the Large Hadron Collider accelerator at CERN (Geneva). It describes the back-end system of the Tile Hadronic Calorimeter of the ATLAS experiment.

Chapter 1 gives an overview of the history and main achievements of the CERN laboratory; it also describes the LHC accelerator and experiments, and with more detail the ATLAS detector. Finally it includes a description of the basic design concepts of the sub-detectors and the Trigger and Data Acquisition system of ATLAS.

The Tile Hadronic Calorimeter (TileCal) sub-detector and in particular the front-end instrumentation and read-out electronics are described in Chapter 2. It also introduces the TileCal calibration systems and the results obtained with them during the first years of operation.

Chapters 3 and 4 describe the Read-Out Drivers (ROD) and the Optical Multiplexer Module (OMB) boards, respectively, which represent the core of the TileCal back-end system. The main components, operating modes and a complete description of the data format processed in the ROD boards are included. Then a description of the OMB module design and the main aspects of its firmware and functionalities is given. The installation, commissioning and the performance of the back-end system during the first three years of ATLAS operation are presented in Chapter 5.

Chapter 6 describes the work done in the implementation of the signal reconstruction algorithms in the Digital Signal Processors (DSP) of the ROD boards. In particular, it explains the formulation of the Optimal Filtering algorithm with a justification of its selection as the

default reconstruction method in TileCal. It includes a description of the implementation of the Optimal Filtering method and the limited precision due to the usage of a fixed point arithmetic in the DSPs.

Chapter 7 is devoted to the study of the performance of the Optimal Filtering implemented in the DSPs. It includes the results of the reconstruction using pseudo-data generated in the laboratory used to corroborate the correct implementation of the method in the DSPs. Then, the performance of the DSP Optimal Filtering implementation is studied using the TileCal Charge Injection System. The last part of the Chapter shows the performance of the reconstruction for physics data, explaining the evolution of the reconstruction strategy as a function of the LHC beam parameters during the first three years of operation. Finally, results about the impact of using the TileCal DSP fixed point arithmetic in the reconstruction of jets in ATLAS are presented.

Jose Alberto Valero Biot

Valencia, June 2014

Contents

Chapter 1

The ATLAS Experiment at the CERN Large Hadron Collider

1.1 CERN

CERN is the European Organization for Nuclear Research. The name CERN derives from the French acronym Conseil European pour la Recherche Nucleaire (European Council for Nuclear Research), which was a provisional council to set-up the laboratory in 1952 with the mandate of establishing a fundamental physics research organization in Europe. The organization became official in 1954 with the title of European Organization for Nuclear Research, although the name CERN was retained. The Globe of Science and Innovation was built on Galileo Galilei's square to celebrate the CERN's 50th anniversary of foundation (Figure 1.1). The Globe is used for general public presentations of science, technology and industry.

The knowledge of the matter in those days was limited to the atomic nucleus, hence the name nuclear. Today, our understanding of matter goes much deeper than the nucleus, and CERN's main area of research is particle physics, the study of the fundamental constituents of matter and the forces acting between them. Because of this, the laboratory operated by CERN is commonly referred to as the European Laboratory for Particle Physics.

Figure 1.1: View of the CERN Globe of Science and Innovation and the ATLAS surface buildings.

CERN is run by 20 European Member States, but many non-European countries are also involved in different ways. Scientists come from around the world to use CERNs facilities.

The discovery of neutral currents in the Gargamel bubble chamber, the first creation of anti hydrogen in the PS20 experiment, the discovery of direct CP violation in the NA48 experiment and the isolation of 38 atoms of antihydrogen are just few examples among a long list of important physics achievements made at CERN. In 1984 Carlo Rubbia and Simon Van der Meer received the Nobel Prize in physics for their contribution in the discovery of the W and Z bosons. This result conformed the unification of the electromagnetic and weak forces, the electroweak theory of the Standard Model. More recently, in 2012 the ATLAS and CMS experiments announced the discovery of a new boson with mass around 125 GeV consistent with long-sought Higgs boson, which led to the nobel prize in physics in 2013 to François Englert and Peter W. Higgs.

The discoveries of the W, Z and the Highs bosons are without a doubt the most important achievements made at CERN since its foundation. Moreover, CERN offers the excellent environment for development of new technologies in other fields like electronics or computing. One example is the World Wide Web (WWW), invented at CERN in 1989, which was conceived

to facilitate the exchange of information between scientists working in different institutes and collaborating through CERN.

1.2 The Large Hadron Collider

In the early times particle accelerators collided high energetic particles with a stationary target where the most valuable energy of the particles was taken up by the target recoil and only a small fraction of it fed the real collision.

Since its foundation people at CERN worked in a new scheme where two particle beams were accelerated and collided with each other. Thus, no recoil energy would be wasted, making for much more efficient collisions. A new Proton Synchrotron (PS) was used to feed two Intersecting Storage Rings (ISR) where two proton beams could be built and finally collide.

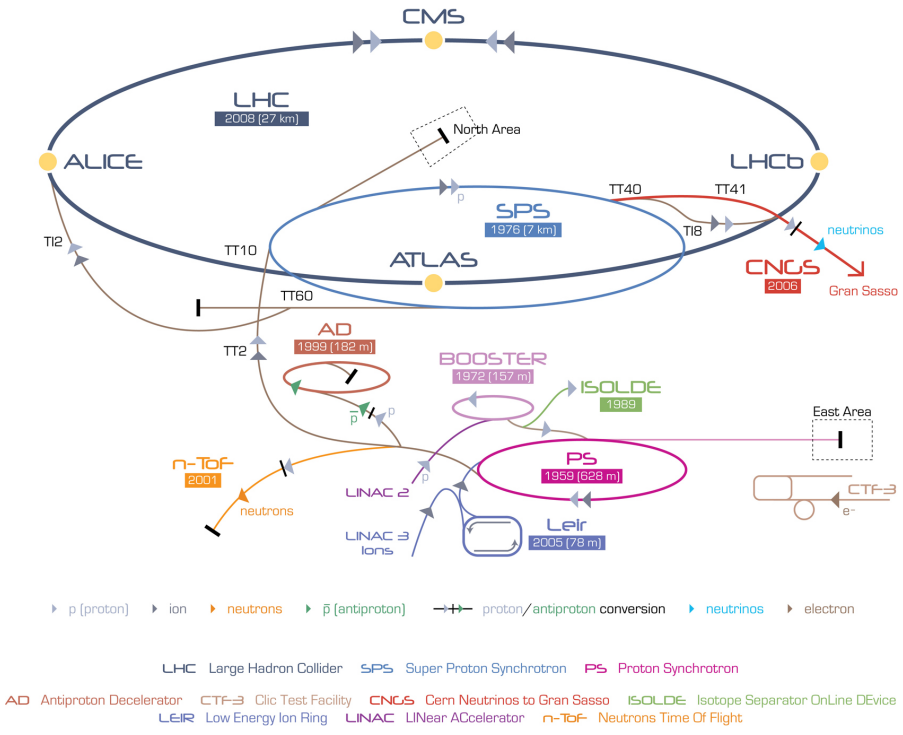


Figure 1.2: The CERN's accelerators complex.

The ISR was commissioned in 1971 and represented the world's first proton-proton collisions. The collider, regarded as a masterpiece, gained CERN the knowledge and expertise for its subsequent colliding beam projects.

In 1979 CERN profited of its ISR investment by deciding to convert its new Super Proton Synchrotron (SPS) into the world's first proton-antiproton collider. The first proton-antiproton collisions were achieved just two years after the project was approved, and in 1983 the W and Z bosons were discovered.

After that the Large Electron-Positron Collider (LEP) was planned, and the ISR was switched off to release resources for its construction. LEP was a circular collider with a circumference of 27 km built in a tunnel straddling the border of Switzerland and France. It was used from 1989 until the end of 2000, when it was shut down and then dismantled in order to make room in the tunnel for the Large Hadron Collider (LHC) [1].

Protons are accelerated and formed in beams in four increasingly large machines before being injected with an energy of 450 GeV into the LHC's 27 km ring (Figure 1.2). The beams are then accelerated in the ring until their energy has increased by a factor of 15, up to 7 TeV. When that energy is reached, the proton beams collide in the center of the experiments. The details of the LHC can be seen in Table 1.1.

Injection energy	450 GeV
Collision energy	7000 GeV
Number of particles per bunch	1.15×10^{11}
Number of bunches per fill	2808
Nominal luminosity	10^{34} cm^{-2} s^{-1}
Inelastic cross section	60 mb
Total cross section	100 mb
Revolution frequency	11.245 kHz
Bunch frequency	40.08 MHz
Circumference length	26.66 km
Radius	4.24 km
Number of dipole magnets	1232
Number of quadrupole magnets	392
Nominal magnetic field strength	8.33 Tesla

Table 1.1: LHC beam parameters.

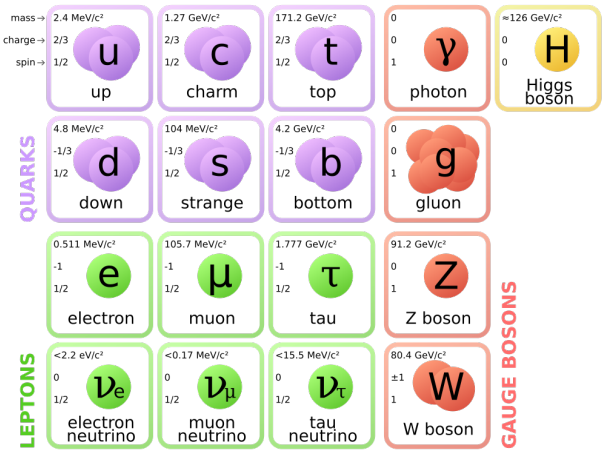


Figure 1.3: Standard model of particle physics.

Four large experiments are installed around the ring of the LHC as shown in Figure 1.2. Two general-purpose experiments, ATLAS (A Toroidal LHC ApparatuS) [2] and CMS (Compact Muon Solenoid) [3], are optimized to study new physics at the TeV scale. The other two experiments are designed to study specific phenomena, LHCb (Large Hadron Collider Beauty) [4] and ALICE (A Large Ion Collider Experiment) [5]. LHCb investigates the CP violation in the bottom quark sector whereas ALICE studies quark-gluon plasma through Pb-Pb and Pb-p collisions.

The most widely used fundamental particle physics theory, the Standard Model, leaves many unsolved questions (Figure 1.3). Among them, the reason why elementary particles have mass, and why are their masses different is one of the most important. The answer may lie within the Brout-Englert-Higgs (BEH) mechanism, which states that particles acquire mass by the interaction with a Higgs field.

Particles which interact strongly with the Higgs field are heavy, whilst those which interact weakly are light. The Higgs field has at least one new particle associated with it, the Higgs boson. After three years of operation, in 2012 both ATLAS and CMS announced the discovery of a new particle with mass around 125 GeV consistent with the Higgs boson (Figure 1.4) [6]. The upcoming years of LHC operation will provide enough statistics to determine the new particle's properties.

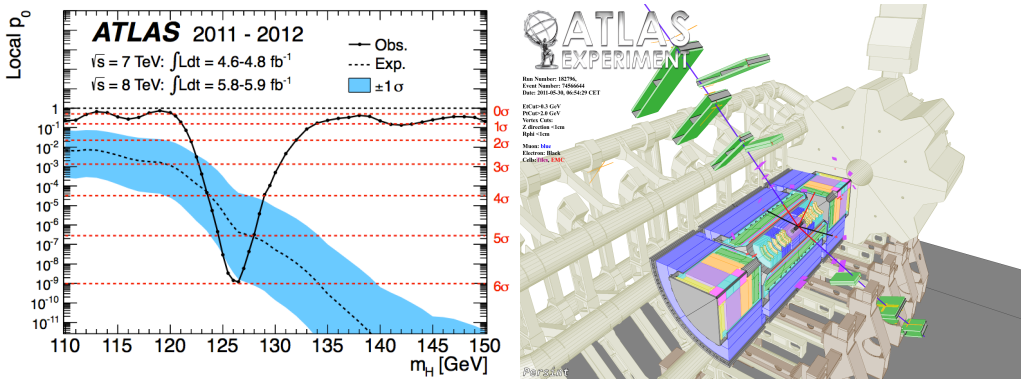


Figure 1.4: Left: The observed (solid) local p_0 as a function of Higgs mass in the low mass range. The dashed curve shows the expected local p_0 under the hypothesis of a SM Higgs boson signal at that mass with its ± 1 σ band. The horizontal dashed lines indicate the p-values corresponding to significances of 1 to 6 σ. Right: Event display of a $2e2\mu$ candidate event with m_{4l}=124.3 GeV. The masses of the lepton pairs are 76.8 GeV and 45.7 GeV.

1.3 The ATLAS Experiment

ATLAS is a general-purpose p-p detector designed to exploit the full discovery potential of the LHC [2] . ATLAS is about 45 meters long, more than 25 meters high and has an overall weight of approximately 7000 tones (Figure 1.5). The inner detector is located in the innermost part of ATLAS. It is built around the beam pipe and is designed especially for tracking and vertexing. It is formed by the Pixel detector, Semiconductor Tracker (SCT) and Transition Radiation Tracker (TRT). It measures the trajectories of the charged particles created in the collisions. The inner detector is embedded in a solenoidal magnet which generates a magnetic field of 2 Tesla. The curvature of the trajectories which results from the the magnetic field bending power, is used to calculate the momentum of the particles. Additionally, the TRT provides electron identification by measurement of the transition radiation photons generated in the radiator material. Electromagnetic and hadronic calorimeters surround the solenoid magnet and are designed to measure the absorbed energy and the direction of the different kinds of particles that cross them. The last layer of the detector is formed by the muon spectrometer and a toroidal magnet. The muon tracking system measures the trajectories of charged particles leaving the calorimeters. The trajectories are bent by the magnetic deflection

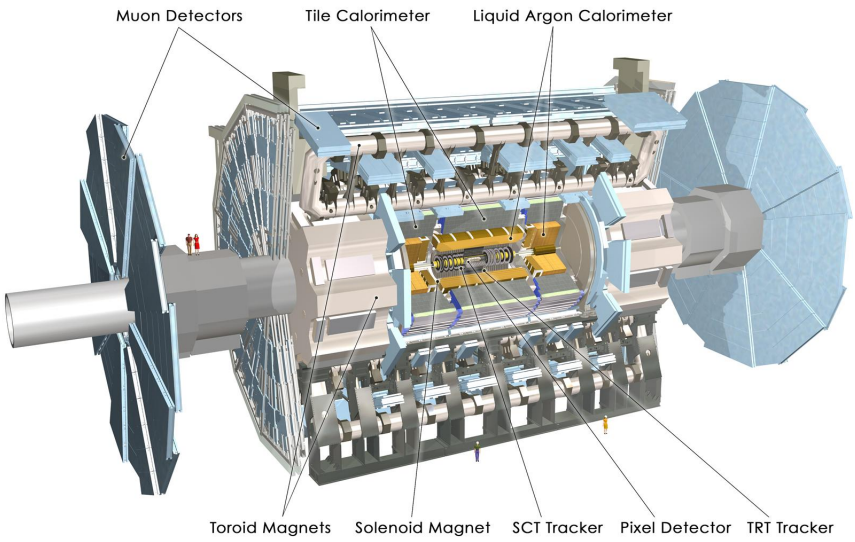


Figure 1.5: The ATLAS detector.

provided by three superconducting air-core toroid magnets, which generate an average field of 0.5 Tesla.

The detector is optimized for a long range of known and hypothetical processes. The observable cross-section for most of the interesting processes is small over a large fraction of the energy range, hence it is an important design requirement to operate at high luminosity and to maximize the detectable rates above the backgrounds by high resolution measurements. The basic design criteria of the detector include the following:

- Very good electromagnetic calorimetry for electron and photon separation and measurement, complemented by a full-coverage hadronic calorimetry for accurate jet and missing transverse energy (E_T^{miss}) measurements.
- High-precision muon measurements, with the capability of guaranteing accurate measurements at the highest luminosity using the external muon spectrometer alone.
- Efficient tracking at high luminosity for high-p_T lepton-momentum measurements, electron and photon identification, τ-lepton and heavy-flavour identification, and full event reconstruction capability at low luminosity.

- Large acceptance in pseudo-rapidity (η) with almost full azimutal angle (ϕ) coverage everywhere. The azimuthal angle is measured around the beam axis. The pseudo-rapidity is measured with respect to the plane perpendicular to the beam and derived from the polar angle (θ):

$$\eta = -ln(\tan\frac{\theta}{2}) \tag{1.1}$$

- Triggering and measurement of particles at low-p_T, providing high efficiencies for most physics processes of interest at LHC.

1.3.1 The Inner Detector

The Inner Detector is designed to reconstruct tracks and decay vertices in any event with high efficiency [7]. Using additional information from the calorimeter and muon systems, the inner detector also contributes to electron, photon, and muon identification, and supplies extra signatures for short-lived particle decay vertices. Important physics considerations for the design of the inner detector are:

- Excellent momentum and impact parameter resolution for tracks with p_T>0.5 GeV up to very high momentum,
- tracking coverage over the range $|\eta|$<2.5,
- high efficiency with equally high noise rejection,
- identification of the charge of high-p_T tracks,
- tagging of jets originating from b-quarks (b-jets),
- reconstruction of soft electrons and secondary vertices from b and τ decays,
- identification of the primary vertex,
- electron identification capability,
- identification of a high-p_T track to reduce the Level 1 electromagnetic cluster trigger rate from jet events.

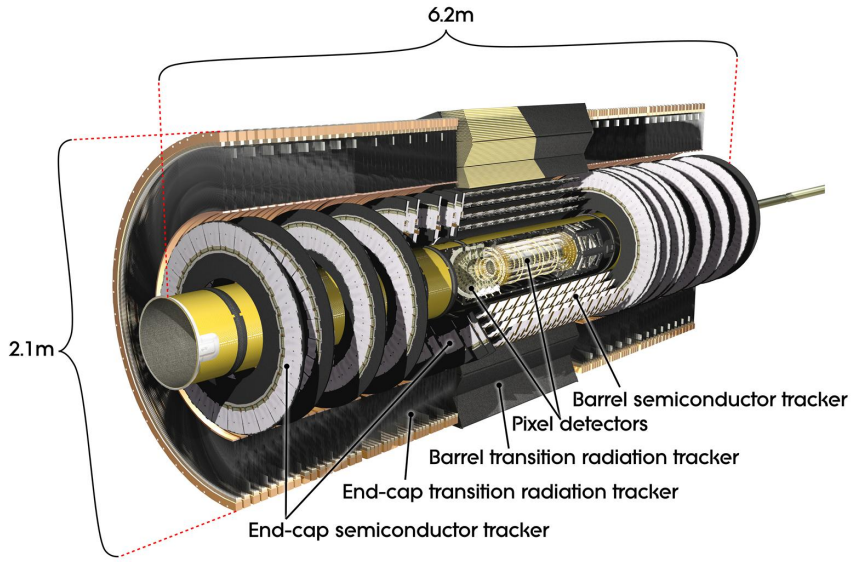


Figure 1.6: The Inner detector.

The magnetic field configuration of the Inner Detector is based on an inner thin superconducting solenoid surrounding the inner detector cavity with a radius of 1.2 m and a length of 5.3 m. It provides an axial magnetic field of 2 Tesla in the centre of the tracking volume. The momentum and vertex resolution requirements from physics call for high-precision measurements to be made with fine granularity detectors, given the very large track density expected at the LHC. The layout of the Inner Detector is shown in Figure 1.6. The outer radius of the Inner Detector cavity is 115 cm. It consists of three units: a barrel section extending over 80 cm, and two identical end-caps covering the rest of the cylindrical part. In the barrel region, high-precision detector layers are arranged on concentric cylinders around the beam axis, while the end-cap detectors are mounted on disks perpendicular to the beam axis. The highest granularity around the vertex region is provided by semi-conductor pixel and strip detectors, the latter employed in the Semiconductor Tracker (SCT). The basic principle of the semiconductor detectors is that the passage of ionizing radiation creates electron-hole pairs in the semiconductor which are collected by an electrical field. The difference between strips and pixels is mainly geometry, pixels being closely spaced pads capable of good two dimensional

reconstruction while strips give a better spacial resolution in one coordinate than the other. The pixel layers are segmented in Rϕ and z, while SCT detector uses small angle (40 mrad) stereo strips to measure both coordinates, with one set of strips in each layer measuring ϕ. The pixel detector is much more radiation tolerant than the silicon strip tracker. The number of layers of the semiconductor detectors must be limited due to the material they introduce and their high cost. A larger number of tracking points is provided by the straw tube tracker also called Transition Radiation Tracker (TRT), which provides continuous tracking with much less material per point and a lower cost. The barrel TRT tubes are parallel to the beam direction. The continuous tracking consists of radial straws arranged into wheels.

1.3.2 Calorimeters

At the LHC about twenty soft collisions per bunch crossing will be produced when operating at design luminosity. Therefore fast detector response and fine granularity are required to minimize the impact of the pileup on the physics performance. The calorimetry part of the ATLAS detector consists of an electromagnetic (EM) calorimeter covering the pseudorapidity region $|\eta|<3.2$, a barrel hadronic calorimeter covering $|\eta|<1.7$, hadronic endcap calorimeters covering $1.4<|\eta|<3.2$, and forward calorimeters covering $3.2<|\eta|<4.8$. The EM calorimeter is a lead/Liquid-Argon (LAr) detector with accordion geometry. The hadronic barrel calorimeter is based on a sampling technique with plastic scintillator plates (tiles) embedded in an steel absorber therefore the name TileCal. At larger rapidities, where higher radiation resistance is needed, the radiation-hard LAr technology is used for all the calorimeters: the Hadronic Endcap Calorimeter (HEC) and the Forward Calorimeter (FCal). A scheme with all the calorimeters for ATLAS can be seen in Figure 1.7.

Electromagnetic Calorimetry

In the barrel, the electromagnetic calorimeter consists of two identical half-barrels covering the rapidity range $|\eta|<1.4$. For each half-barrel (divided into 16 modules) the calorimeter is made of 1024 accordion-shaped absorbers alternating with 1024 readout electrodes, arranged with a complete ϕ symmetry around the beam axis. Between each pair of absorbers, there are two liquid argon gaps, separated by a readout electrode.

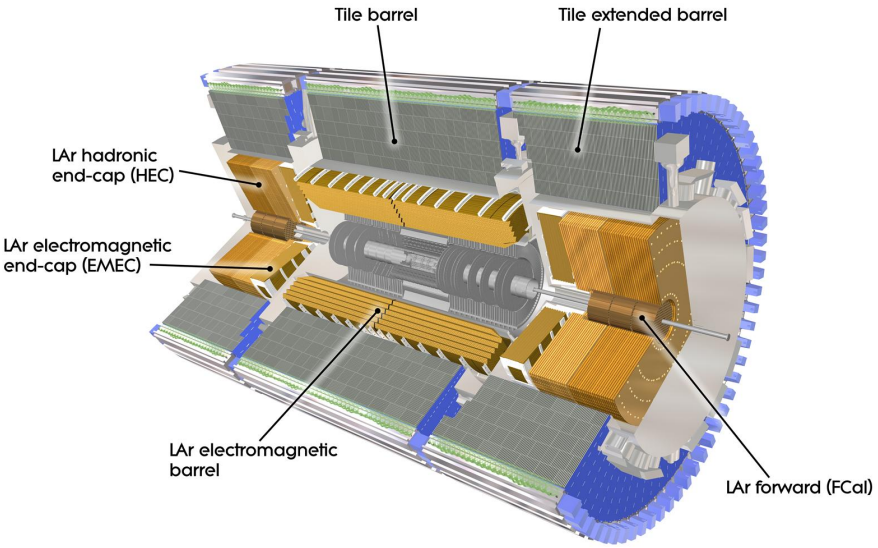


Figure 1.7: ATLAS calorimeters system.

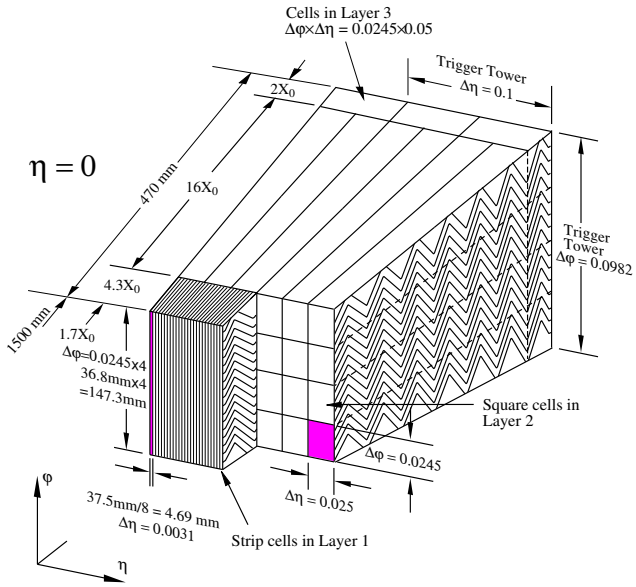


Figure 1.8: Diagram of a LAr EM calorimeter barrel module. It is shown the longitudinal segmentation, the cell size and the accordion structure.

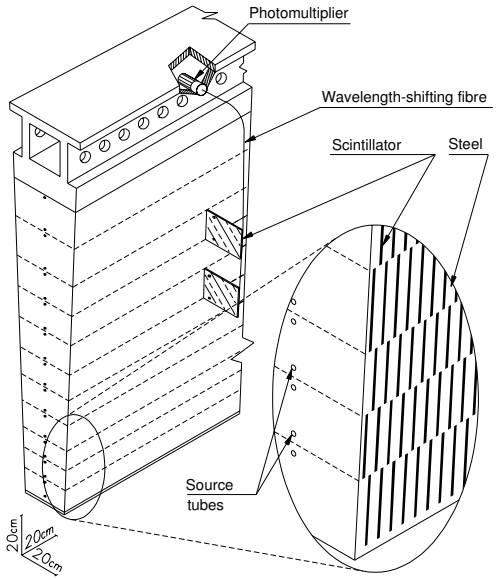


Figure 1.9: TileCal module components and structure.

The ElectroMagnetic EndCap (EMEC), the HEC and the FCal calorimeters are placed inside the endcap cryostat. The EMEC, which covers the range $1.375<|\eta|<3.2$, uses the same technique as in the barrel part. The HEC covers the range $1.5<|\eta|<3.2$ and uses copper plates as absorbers, with parallel geometry in this case. The FCAL placed in the $3.2<|\eta|<4.9$ region, provides EM coverage as well as hadronic showers by using copper and tungsten as absorbers, respectively. The EM calorimeter is segmented in three longitudinal samplings in the $|\eta|<2.5$ region and in two samples in the $|\eta|>2.5$ region, as Figure 1.8 shows. The total thickness of the EM calorimeter is above 24 radiation lengths for the barrel and above 26 for the endcaps.

Hadronic Calorimetry

Hadronic calorimetry at the LHC is mainly designed to identify the energy and direction of jets and to measure the E_T^{miss} [8] . Fragmentation, gluon radiation and the presence of magnetic fields are intrinsic effects that limit the resolution of these measurements. However, at the LHC designed luminosity, the pile-up energy from minimum-bias events also becomes important. In order to be sensitive to interesting physics, an energy resolution of

$50\%/\sqrt{E}\oplus 3\%$ and segmentation of $\Delta\eta$ x $\Delta\phi$=0.1 x 0.1 is required for the central region. For the forward region, a resolution of $100\%/\sqrt{E}\oplus 10\%$ and segmentation of $\Delta\eta$ x $\Delta\phi$=0.2 x 0.2 is sufficient. The ATLAS hadronic calorimetry covers a pseudo-rapidity range $|\eta|<5$. Different techniques are used depending on η. The Tile calorimeter covers $|\eta|<1.7$. It is composed by a barrel and two extended barrels made out of plastic scintillating tiles as active medium. For $1.5<|\eta|<4.9$ liquid-argon techniques are used for the two end-caps ($1.5<|\eta|<3.2$) and the two high density forward calorimeters ($3.2<|\eta|<4.9$). Both the HEC and FCal are embedded into the same cryostat as the electromagnetic end-cap calorimeters. In the TileCal the light produced in the scintillating tiles is collected through wavelength shifting fibers and guided to a photomultiplier (see Figure 1.9). Readout cells are defined by grouping together several fibers to the same photomultiplier, achieving three longitudinal samplings with a segmentation of $\Delta\eta$ x $\Delta\phi$=0.1 x 0.1 for the two inner layers and (0.2 x 0.1) for the outer layer. The analogue electrical signal of the photomultiplier is digitized in samples of 10 bits taken each 25 ns. These data are sent to the ReadOut Driver (ROD) system where online reconstruction algorithms are applied. The TileCal and the ROD system are described in detail in Chapter 2.

The hadronic end-caps are made up of two equal diameter wheels. The first wheel is built out of 25 mm copper plates as absorber and the second wheel uses 50 mm copper plates. Compared to lead, copper has a shorter interaction length that allows to increase the size of liquid-argon gaps between plates, thereby reducing the electronic noise, the integration time and pile-up noise. In both wheels the absorber plates are separated by 8.5 mm gaps filled with liquid-argon and a structure of three electrodes that divide the gap into four drift spaces of $\sim$1.8 mm.

The forward calorimetry should be efficient at forward jet tagging and E_T^{miss} reconstruction. The forward calorimeters are high density detectors in order to accommodate at least 9 interaction lengths of active material in rather short longitudinal space. Each forward calorimeter is divided into three longitudinal sections. In the first section the absorber is copper while in the second and third sections is tungsten. The calorimeter consists of a metal matrix (the absorber) filled with rods (electrodes). Liquid-argon is the active medium and fills the gaps between the matrix and the rods.

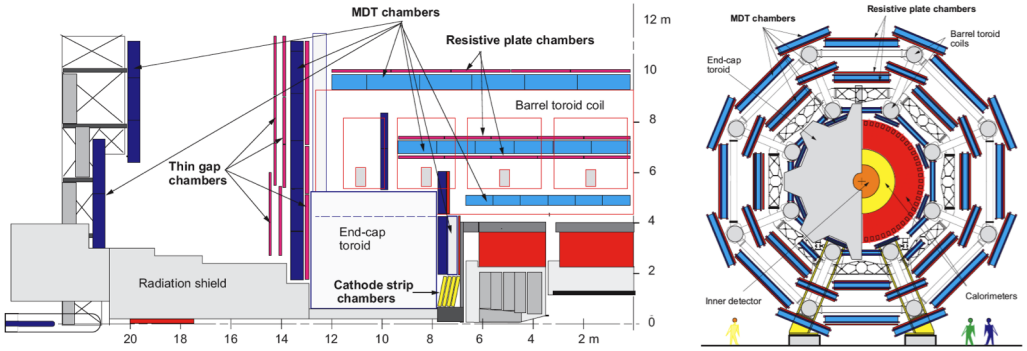


Figure 1.10: The ATLAS muon spectrometer in the rz (left) and xy view (right).

1.3.3 The Muon Spectrometer

The ATLAS muon spectrometer has been designed to make efficient use of the magnet bending power with a coverage of $|\eta|<3$. It provides projective towers in η and ϕ and is made out of practical chamber dimensions for production, transport and installation [9]. Figure 1.10 shows the position of the muon chambers. The spectrometer is divided into three regions, barrel region ($|\eta|<1.05$), transition region ($1.05<|\eta|<1.4$) and endcap region ($|\eta|>1.4$). Four different technologies have been used depending on spatial and timing resolution, resistance to radiation and engineering considerations: Monitored Drift Tube chambers (MDT), Cathode Strip Chambers (CSC), Resistive Plate Chambers (RPC) and Thin Gap Chambers (TGC).

The MDT chambers are composed of multilayers of high-pressure drift tubes. Each multilayer is mounted on each side of the support structure. The drift tubes are made of aluminum, 30 mm of diameter, with a central wire of W-Re. They work at 3 bar absolute pressure with a non-flammable mixture of Ar-CO2.

The CSCs are multiwire proportional chambers operated with a mixture of $Ar - CO_2 - CF_4$. The distance between anode wires (2.5 mm) equals the distance to the cathode. The cathode readout is segmented into strips (5.08 mm) orthogonal to the anode wires. The precision coordinate is obtained by measuring the induced avalanche in the segmented cathode, achieving space resolutions better than 60 μm. The RPC is a gaseous parallel-plate detector with a typical space-time resolution of 1 cm x 1 ns with digital readout. It is composed by two

parallel resistive plates made out of bakelite. The plates are separated by spacers that define the size of the gas gaps. The gas is a mixture of $C_2H_2F_4$. A uniform electric field of a few kV/mm produces the avalanche multiplication of ionization electrons. The signal is read out via capacitative coupling to metal strips placed at both sides of the detector and grounded.

The TGC is built with 50 μm wires separated by 2 mm. The wires are placed between two graphite cathodes at a distance of 1.6 mm. Behind the graphite cathodes, strips or pads are located to perform a capacitive readout in any desired geometry. Some advantages of these chambers are a fast signal, typical rise time 10 ns and low sensitivity to mechanical deformations.

In the barrel region the chambers are situated in three concentric cylinders (so called stations) around the beam axis at a radial distance of 4.5 m, 7 m and 10 m. MDT chambers are used for high precision measurements and RPC for triggering. The low-p_T muon trigger uses two double-layer RPCs located on each side of the middle station, while the high-p_T trigger uses one triple layer chamber located at the outer barrel muon station. In the transition and end-cap region most of the chambers are installed perpendicular to the beam axis (see Figure 1.10). In the transition region ($1.05<|\eta|<1.4$) the muon track is measured with three vertical stations, placed inside or near the barrel magnet. In the end-cap region ($|\eta|>1.4$), the stations are located before and after the end-cap toroid magnets and a third one near the cavern wall. The trigger is provided by the TGC chambers while precision measurements are provided by the MDT chambers at small η and the CSC chambers at large rapidity.

1.3.4 Magnetic Field

The ATLAS magnetic field is optimized to increase the identification power of the sub-detectors in a light and open structure which minimizes scattering effects [10]. It consists of a central solenoid servicing the inner detector with an axial magnetic field, surrounded by eight large scale air-core coils generating a toroidal magnetic field for the muon spectrometer (Figure 1.11). The Nb-Ti superconductor in a copper matrix technology is used in this case. The magnet system weights 1300 tons and is cooled by liquid He at 4.5 K.

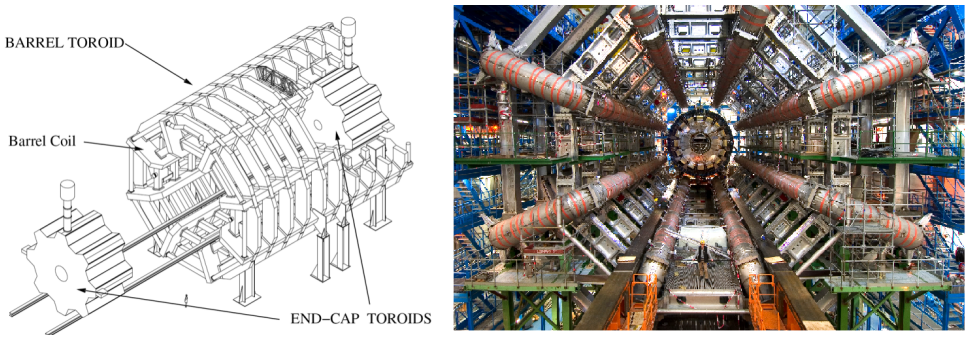


Figure 1.11: Sketch of the ATLAS superconducting air-core toroid magnet system (left) and picture of the barrel toroid (right).

1.4 The ATLAS Trigger and Data Acquisition System

The ATLAS trigger system consists of three levels of event selection: Level 1, Level 2, and Event Filter (EF) [11]. The Level 1 trigger is implemented using custom-made electronics, while the Level 2 and EF is almost entirely based on commercially available computers and networking hardware. The Level 2 and EF together form the High-Level Trigger (HLT). A block diagram of the trigger and data acquisition systems is shown in Figure 1.12. The Level 1 trigger searches for signatures from high-p_T muons, electrons/photons, jets, and τ-leptons decaying into hadrons. It also selects events with large E_T^{miss} and large total transverse energy. The Level 1 trigger uses reduced-granularity information from the RPCs and TGCs for high-p_T muons, and the calorimeter for the rest of signatures. The maximum Level 1 acceptance rate that the detector readout systems can handle is 75 kHz (upgradeable to 100 kHz), and the Level 1 decision must reach the front-end electronics within 2.5 μs after the bunch crossing with which it is associated. The Level 2 trigger is seeded by Regions-of-Interest (RoIs). These are regions of the detector where the Level 1 trigger has identified possible trigger objects within the event. The Level 2 trigger uses the RoIs information on coordinates, energy, and type of signature to limit the amount of data which must be transferred out from the detector readout. The Level 2 trigger reduces the event rate to below 3.5 kHz, with an average event processing time of approximately 40 ms. The EF uses offline analysis procedures on fully-built events to further reduce the event selection to a rate of up to 800 Hz,

with an average event processing time of the order of four seconds. HLT algorithms use full granularity and precision of the calorimeter and muon chamber data, as well as the data from the inner detector, to refine the trigger selections. Better energy resolution improves the threshold cuts, while track reconstruction in the Inner Detector significantly enhances the particle identification (for example distinguishing between electrons and photons). The event selection at both Level 1 and Level 2 primarily uses inclusive criteria, for example high-E_T objects above defined thresholds. One exception is the Level 2 selection of events containing the decay of a B-hadron, which requires the reconstruction of exclusive decays into particles with low momentum.

The Data AcQuisition system (DAQ) receives and buffers the event data from the detector specific readout electronics at the Level 1 trigger rate through point-to-point ReadOut Links (ROLs). It transmits to the Level 2 farm any data requested by the trigger and, for those events fulfilling the Level 2 selection criteria, event-building is performed. The assembled events are input to the EF, and the events selected there are moved to permanent event storage. In addition to controlling movement of data down the trigger selection chain, the DAQ system also provides infrastructure for the configuration, control and monitoring of the ATLAS detector during data-taking. Supervision of the detector hardware (gas systems, power-supply voltages, etc.) is provided by the Detector Control System (DCS) [12].

1.4.1 The First Level of Trigger

The first level of trigger (Level 1) performs the initial event selection based on information from the calorimeters and muon detectors as shown in Figure 1.13. The calorimeter selection is based on reduced granularity and precision information from all the calorimeters gathered by the Level 1 Calorimeter (L1Calo). The triggers select objects with high p_T and large E_T^{miss} and scalar $\sum E_T$. The trigger signatures can require isolation where the energetic particle must have a minimum angular separation from any significant energy deposit in the same trigger. The information for each bunch crossing used in the Level 1 trigger decision is the multiplicity of hits for 4 to 16 programmable E_T thresholds per object type.

The Level 1 muon trigger is based on signals in the muon trigger chambers: RPCs in the barrel and TGCs in the end-caps. The trigger searches for patterns of hits consistent with

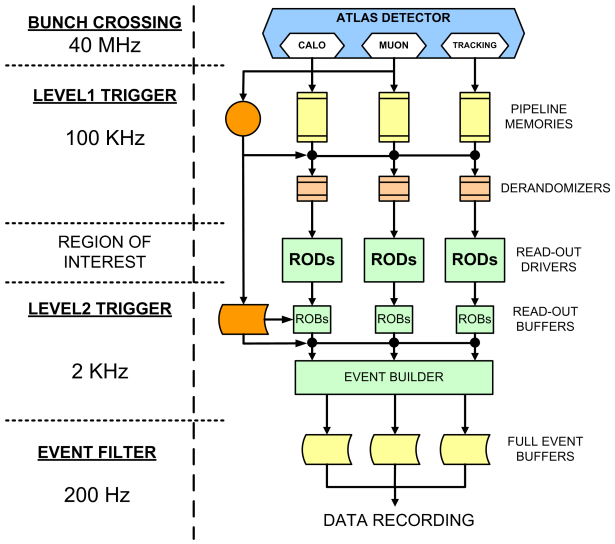


Figure 1.12: ATLAS data acquisition system and trigger levels.

high-p_T muons originating from the interaction region. The logic provides six independently-programmable p_T thresholds. The information for each bunch crossing used in the Level 1 trigger decision is the multiplicity of muons for each of the p_T thresholds. Muons are not double-counted across the different thresholds.

The overall Level 1 accept decision is made by the Central Trigger Processor (CTP), which combines the information for different object types. Trigger menus can be programmed with up to 256 distinct items, each item being a combination of requirements on the input data. The trigger decision, together with the LHC clock and other signals, are distributed to the detector front-end and readout systems via the Timing, Trigger and Control (TTC) system, using an optical-broadcast network [13].

While the Level 1 trigger decision is based only on the multiplicity of trigger objects or flags indicating which thresholds were passed, for global quantities, information about the geometric location of trigger objects is retained in the muon and calorimeter trigger processors. Upon the event being accepted by the Level 1 trigger, this information is sent as RoIs to the Level 2 trigger, where it is used as a seed for the selection performed by the HLT. An essential function of the Level 1 trigger is unambiguous identification of the bunch crossing of interest.

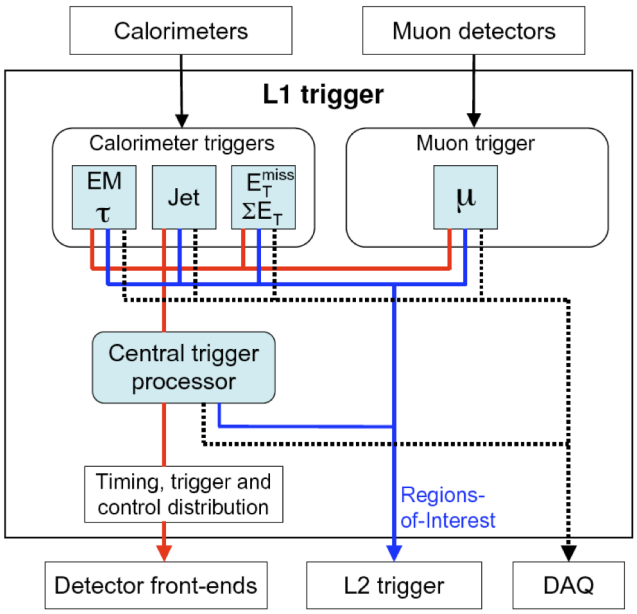


Figure 1.13: Block diagram of the Level 1 trigger.

The very short (25 ns) bunch crossing interval makes this a challenging task. In the case of the muon trigger, the physical size of the muon spectrometer implies times-of-flight exceeding the bunch crossing interval. For the calorimeter trigger, a serious complication is that the width of the calorimeter signals extends over many bunch crossings. While the trigger decision is being taken, the data from all detector channels has to be retained in pipeline memories. These memories are contained in custom electronics placed on or near the detector, where often radiation levels are high and access is difficult. In the interest of cost and reliability, it is desirable to keep the pipeline length as short as possible. The Level 1 latency, which is the time from the proton-proton collision until the Level 1 trigger decision, must therefore be kept as short as possible. The design of the trigger and front-end systems requires the Level 1 latency to be less than 2.5 μs, with a target latency of 2.0 μs, leaving 0.5 μs contingency. About 1 μs of this time is accounted for by cable-propagation delays alone. To achieve this aim, the Level 1 trigger is implemented as a system of purpose-built hardware processors.

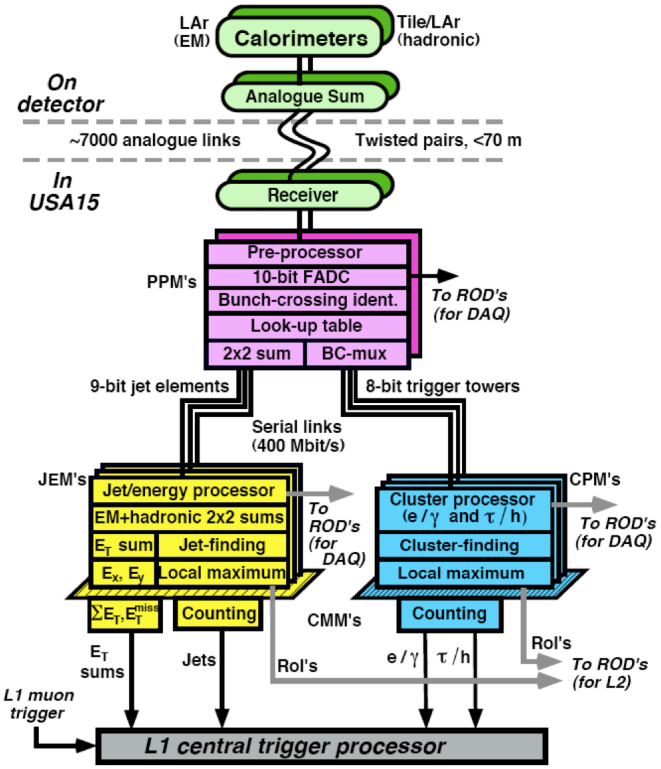


Figure 1.14: Architecture of the Level 1 calorimeter trigger. Analogue data from the calorimeters are digitized and associated with the correct bunch crossing in the pre-processor and then sent to two algorithmic processors, the Jet/Energy-Sum (JEM) processor and the Cluster Processor (CPM). The resulting hit counts and energy sums are sent to the Central Trigger Processor.

Calorimeter Trigger

The L1Calo is a pipelined digital system designed to work with about 7000 analogue trigger towers of reduced granularity (0.1 x 0.1 in $\Delta\eta$ x $\Delta\phi$ in most parts, but larger at higher $|\eta|$) from the electromagnetic and hadronic calorimeters. It sends the results for each LHC bunch crossing to the CTP approximately 1.5 μs after the event occurs, resulting in a total latency for the L1Calo chain of about 2.1 μs, well within the allowed envelope. The complete diagram of the L1Calo is presented in Figure 1.14.

The L1Calo system is located off-detector in the service cavern USA15. Its architecture,

shown in Figure 1.14, consists of three main sub-systems. A pre-processor digitizes the analogue input signals, then uses a digital filter to associate them with a specific Bunch Crossings that uses a Look-Up Table to produce the transverse-energy values used for the trigger algorithms. Finally the Cluster Processor (CP) and Jet/Energy-sum Processor (JEP) subsystems receive the data in parallel. The CP sub-system identifies electron/photon and τ-lepton candidates with E_T above the corresponding programmable threshold and satisfying, if required, certain isolation criteria. The JEP receives jet trigger elements, which are 0.2 x 0.2 sums in $\Delta\eta$ x $\Delta\phi$, and uses these to identify jets and to produce global sums of scalar and missing transverse energy. Both processors count the multiplicities of the different types of trigger objects. The CP and JEP send these multiplicities, as well as the transverse-energy threshold information, to the CTP for every bunch crossing. When there is a Level 1 accept (L1A) decision from the CTP, the stored data from the L1Calo are read out by the DAQ: this includes input data, intermediate calculations and trigger results in order to allow full monitoring and verification of the Level 1 trigger functionality. These data can also provide useful diagnostics for the LHC machine and the ATLAS sub-detectors. The types and positions of jet, τ-lepton and electromagnetic cluster candidates are also collected and sent to the RoI builder for use by the Level 2 trigger. The L1Calo architecture is relatively compact, with a minimal number of crates and cable links. This helps in reducing the latency. Some of the hardware modules were designed to fulfill several different roles in the system, in order to reduce hardware costs and design efforts, as well as to reduce the number of spares required.

Muon Trigger

The Level 1 muon trigger is based on finely segmented detectors (the RPCs in the barrel and the TGCs in the end-caps) with a sufficient timing accuracy to provide unambiguous identification of the bunch crossing containing the muon candidate. The trigger in both the barrel and the end-cap regions is based on three trigger stations each (Figure 1.15). The basic principle of the algorithm is to require a coincidence of hits in the different trigger stations within the path of a muon from the interaction point through the detector. The width of the path is related to the p_T threshold to be applied. A system of programmable coincidence logic boards allow concurrent operation with a total of six thresholds, three associated with

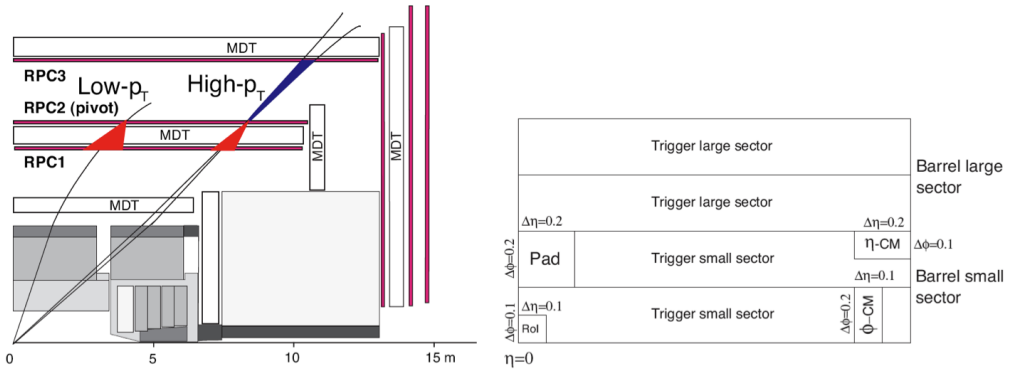


Figure 1.15: Right: Schema of the Level 1 muon barrel trigger. Left: Segmentation of the Level 1 muon barrel trigger. The RPCs are arranged in three stations: RPC1, RPC2, and RPC3.The low-p_T and high-p_T roads are also shown.

the low-p_T trigger (threshold range approximately 6-10 GeV) and three associated with the high-p_T trigger (threshold range approximately 8-35 GeV).

1.4.2 Data Acquisition System and High-Level Trigger

The main components of the Data AcQuisition system and High-Level Trigger (DAQ/HLT) are: readout, Level 2 trigger, Event-Building, Event Filter, configuration, control and monitoring (Figure 1.16) [14]. The movement of events from the detector to mass storage commences with the selection of events by the Level 1 trigger. During the latency of the Level 1 trigger selection, up to 2.5 μs, the event data are buffered in memories located within the detector-specific front-end electronics. On selection by the Level 1 trigger the event data is transferred to the DAQ/HLT system over dedicated ReadOut Links (ROLs), having first transited through the detector-specific RODs. The event fragments are received into their respective ReadOut Buffers (ROBs) contained in the ROS units where they are temporarily stored and provided, on request, to the subsequent stages of the DAQ/HLT system. For every selected event, the Level 1 trigger sub-systems (calorimeter, muon, and CTP) also provide the RoI information on eight ROLs, a dedicated data path to the RoI builder where it is assembled into a single data structure and forwarded to one of the Level 2 SuperVisor (L2SV). As its name suggests, the L2SV marshals the events within the Level 2 trigger. It receives the RoIs

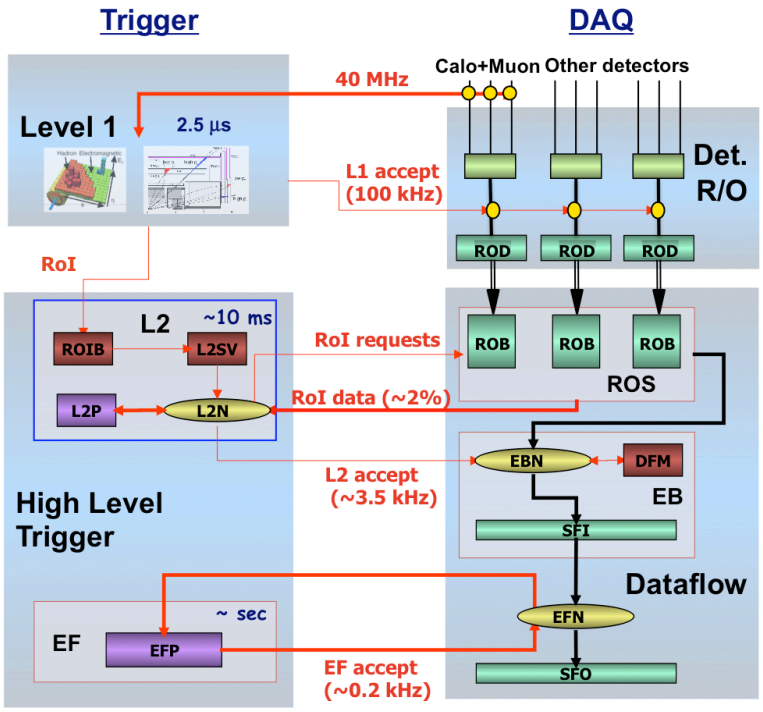


Figure 1.16: Schema of the ATLAS Trigger and Data Acquisition system. The Level 1 trigger system receives the data directly from the front-end electronics of the muon and calorimeter sub-systems. The data reconstructed by the RODs is transferred to the HLT trigger system for all the sub-systems.

and assigns each event to one of the Level 2 Trigger Processing Units (L2PUs) for analysis.

The L2PU, based on the RoI information, requests for event data are made to the appropriate ROS. The sequence of data requests is determined by the type of RoI identified by the Level 1 trigger and the configuration of the Level 2 trigger processing. The result, accept or reject, of the analysis is returned to the L2SV which subsequently forwards it to the Data-Flow Manager (DFM). In addition it also sends a summary of the analysis performed to a Level 2 trigger specific ROS.

The DFM marshals the events during the event-building. For those events which were found not to fulfill any of the Level 2 selection criteria, the DFM informs all the ROSs to expunge the associated event data from their respective ROBs. Each event which has been

selected by the Level 2 trigger is assigned by the DFM to a Sub-Farm Input (SFI) in an event-building node. The SFI collects the event data from the ROSs and builds a single event-data structure, the event. An SFI can build more than one event at a time and the requests to the ROSs for their data are dispatched according to an algorithm which ensures the quantity of data being received by the SFI does not exceed its available input bandwidth. The full event structure is sent to the Event Filter for further analysis. On completing the building of an event an SFI notifies the DFM, which subsequently informs all the ROSs to expunge the associated event data from their respective ROBes. The event filter, in addition to the selection, classifies the selected events according to a predetermined set of event streams and the result of this classification is added to the event structure. Selected events are subsequently sent to a Sub-Farm Output (SFO) in the output nodes of the DAQ/HLT system. Conversely, those events not fulfilling any of the event filter selection criteria are expunged from the system. The events received by an SFO are stored in its local file system according to the classification performed by the event filter. The event files are subsequently transferred to CERN's central data-recording facility. Figure 1.17 show the trigger rates as a function of time for a typical 2012 run for Level 1, Level 2 and Event Filter selections. The rates follow the evolution of the peak luminosity of the LHC fill.

ReadOut System

The ROS receives event data from the detector RODs via 1574 ROLs. All the ROLs have the same design and implementation, based on the S-link interface that allows the transmission of 32-bit data at 40.08 MHz. Thus, corresponds to 160 Mbyte/s with flow control and error detection. ROBs are located at the receiving end of the ROLs, there being one ROL associated to one ROB. Three ROBs are physically implemented on a ROB Input card (ROBin). The ROS, which is implemented on a server-class PC can hold up to 6 ROBin cards. The ROS provides multiplexing of up to 18 ROLs to the subsequent components of the DAQ/HLT. A request by an L2PU for data involves, on average, one or two ROBs per ROS, whereas the requests for data from the event building nodes concern the event data from all the ROBs of a ROS. In either case, the ROS replies to the requester with a single data structure. At the Level 1 trigger rate of 75 kHz, and an average of 1 kbyte received per ROL, the ROS is able

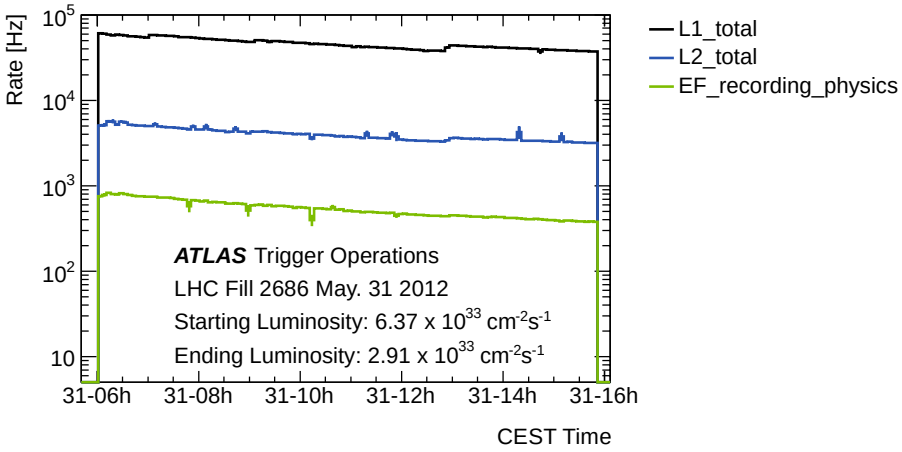


Figure 1.17: Data trigger output rate and recording rates (Hz) as a function of time for the LHC fill 2686 in 2012.

to concurrently service up to approximately 20 kHz of data requests from the Level 2 trigger, up to 3.5 kHz of requests from event building nodes, and expunge events on request from the DFM. The rate of data requests received by a specific ROS depends on the $\Delta\eta - \Delta\phi$ region of the data it receives over the ROLs and from which detector it receives data. For example, a ROS which receives data from the EM Barrel region is solicited for data more frequently than a ROS associated with the barrel MDTs.

Level 2 Trigger

The Level 2 trigger is achieved by the combined functionality of the RoI builder, L2SV, L2PU and Level 2 trigger-specific ROS. The RoI builder receives the RoI information from the different sources at the Level 1 trigger rate on eight input ROLs and merges them into a single data structure. The single data structure containing the RoI data is transmitted by the RoI builder over one of the output ROLs to the L2SVs. L2SVs marshal the events through the Level 2 trigger. The principal component of the Level 2 trigger is the Level 2 processing farm, where the event selection is executed. The system is designed to provide an event rejection factor of about 30, with an average throughput per farm node of about 200 Hz, using the data

located in the RoIs, i.e. 12% of the full data of an event. The number of L2PU applications performing the physics selection per node is configurable. On the hardware currently deployed there are eight L2PUs per node, and one L2PU per processing core of the node. At the end of its event analysis, the L2PU sends to the trigger specific ROS information which summarizes the results of its analysis. Subsequently, this ROS participates in event building like any other ROS within the system, its event data being the Level 2 triggers summary analysis. In this way, the results of the Level 2 triggers analysis are built into the final event and subsequently used by the event filter to seed its selection. The failure of one or more L2PUs during run time does not incur system down time. The system continues to operate at a reduced rate while the failed application, the L2PU, can be restarted under the supervision of the run control.

Event-Building

The event-building functionality is provided by the DFM, ROSs and SFIs. The SFI is the application which collects the event data from the ROSs and assembles the event as a single formatted data structure. An SFI is configured with a randomized list of the ROSs within the system, which is used to define the order in which data requests are sent to the ROSs. This results in the randomization of the traffic pattern in the underlying network and hence improved network performance. To meet the rate requirements a number of SFIs work in parallel, each instance building a number of events concurrently. Each SFI informs the DFM of its readiness to receive events, and the DFM allocates events to the SFIs so as to ensure that the load is balanced across all available SFIs.

Event Filter

The Event Filter is a processing farm; on each processing node a configurable number of independent processing tasks receive and process events. Unlike the Level 2 trigger, these tasks are based on standard ATLAS event reconstruction and analysis applications. The steering of the event selection is the same as Level 2. For those events passing the selection criteria, a subset of the data generated during the event analysis is appended to the event data structure, enabling subsequent offline analysis to be seeded by the results from the event filter. An integral part of the selection process is the classification of the events according

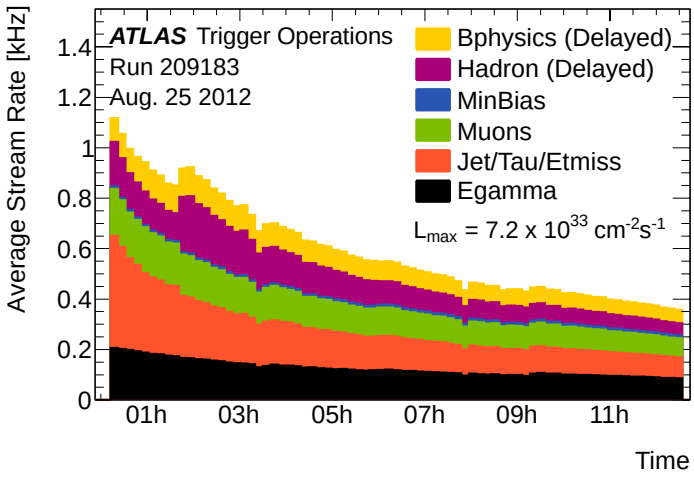


Figure 1.18: Event Filter stream recording rates (kHz) from ATLAS run 209183 (2012) with a peak luminosity of 7.2×10^{33} cm^{-2}s^{-1}. Filtered for LHC stable beams and ATLAS ready. The x-axis has an arbitrary offset.

to the ATLAS physics streams. Figure 1.18 shows the recording rates as a function of time for each stream for a typical 2012 run. The rates diminish with the decreasing instantaneous luminosity.

Event Output

The main functionality of the event-filter SFOs is to receive events which have passed the event filter selection criteria, interface the DAQ/HLT to CERN's central data-recording facility, and de-couple the data-taking process from possible variations in the central data-recording service. The SFO maintains, locally, a set of files into which it records events at a peak event rate of up to 400 Hz and stores them up to 24 hours. Under normal operating conditions, this storage capacity is only partially used. The set of files maps to the ATLAS-defined data streams: electrons, muons, jets, photons, E_T^{miss} and τ-leptons, and B-physics. Each event is recorded in one or more files according to the stream classification made by the Event Filter processing task.

In addition to the data streams mentioned above, a subset of the events is also written to

calibration streams and to the express stream. The express stream is a subset of the events selected by the event filter that represent approximately the 10% of the data randomized that is useful for fast monitoring purposes. The calibration streams provide the minimum amount of information needed for detector calibration seeded by dedicated Level 1 triggers.

Chapter 2

The ATLAS Tile Hadronic Calorimeter

2.1 Detector Overview

The TileCal detector is a sampling calorimeter using steel as the absorber and scintillator tiles as the active medium. It covers the region, $|\eta|<1.7$, behind the liquid argon electromagnetic calorimeter and is subdivided into a central long barrel, 5.8 m in length, and two extended barrels, 2.6 m in length each with an inner radius of 2.28 m and an outer radius of 4.25 m, as shown in Figure 2.1. The radial depth of the Tile calorimeter is approximately 7.4 λ (interaction lengths). Each barrel is azimutally divided into 64 modules or wedges of size $\Delta\phi \sim 0.1$ with a weight of 3000 tonnes [15]. The assembled module forms an almost-periodic steel-scintillator structure with a ratio by volume of approximately 4.7:1. The geometry is sketched in Figure 2.2. The orientation of the scintillator tiles are radially disposed and normal to the beam line. This, in combination with wavelength-shifting (WLS) fibers readout coupled to the tile edges, allows almost seamless azimuthal calorimeter coverage. The grouping of the readout fibers into bundles define the calorimetric cells which are read out by PhotoMultiplier Tubes (PMTs) and also provides an approximate projective geometry in pseudo-rapidity. The gap region between the barrel and the extended barrel is instrumented with special cells, made

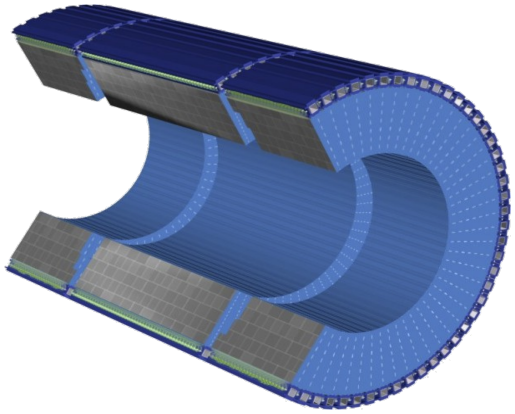


Figure 2.1: Three barrels of TileCal.

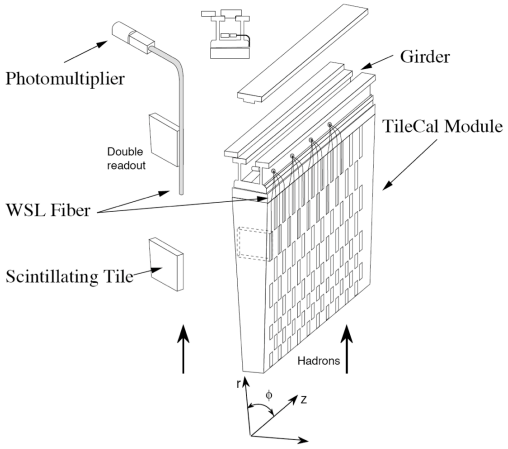


Figure 2.2: Schematic of a TileCal module and main components.

of steel-scintillator sandwiches with the same sampling fraction as the rest of the calorimeter and with thin scintillator counters in the sectors where the available space in the gaps is even more limited. The electronics and readout of the tile calorimeter are highly integrated with the mechanical structure. The photomultiplier tubes and all the front-end electronics are mounted on aluminium units, called superdrawers, which are inserted inside the supporting girder at the rear of each module. The front-end electronics also provide analogue sums from subsets of channels, forming trigger towers, for the Level 1 trigger. The low-voltage power supplies which power the front-end electronics are mounted in an external steel box, which has the cross-section of the girder and which also contains the external connections for power and other services. Finally, the calorimeter is equipped with three calibration systems: charge injection, laser and a ^{137}Cs radioactive source that are used to calibrate the signals to the electromagnetic scale with very good precision.

2.2 Mechanical Structure

The mechanical structure of TileCal is designed as a self-supporting, segmented structure comprising 64 modules, each subtending 5.625 degrees in azimuth, for each of the three barrels.

The module sub-assembly is shown in Figure 2.2. Each module contains a precision-machined strong-back steel girder, the edges of which are used to establish a module-to-module gap of 1.5 mm at the inner radius. To maximize the use of radial space, the girder provides both the volume in which the readout electronics are contained and the flux return for the solenoid field. The readout fibers, suitably bundled, penetrate the edges of the girders through machined holes, into which plastic rings have been precisely mounted. These rings are matched to the position of photomultipliers. The fundamental element of the absorber structure consists of a 5 mm thick master plate, onto which 4 mm thick spacer plates are glued in a staggered fashion to form the pockets in which the scintillating tiles are located. The master plate was fabricated by high-precision die stamping to obtain the dimensional tolerances required to meet the specification for the module-to-module gap. At the module edges, the spacer plates are aligned into recessed slots, in which the readout fibers run. Holes in the master and spacer plates allow the insertion of stainless-steel tubes for the radioactive source calibration system. Each module is constructed by gluing the structures described above into sub-modules on a custom stacking fixture. These are then bolted onto the girder to form modules, with care being taken to ensure that the azimuthal alignment meets the specifications. The calorimeter is assembled by mounting and bolting modules to each other in sequence. Shims are inserted at the inner and outer radius load-bearing surfaces to control the overall geometry and yield a nominal module-to-module azimuthal gap of 1.5 mm and a radial envelope which is generally within 5 mm of the nominal one. The TileCal Valencia-IFIC group built 50% of the TileCal modules for one of the extended barrels of the detector.

2.2.1 Optical Components

Eleven sizes of scintillating tiles (one for each depth in radius) of 3 mm thickness and with radial lengths ranging from 97 mm to 187 mm and azimuthal lengths ranging from 200 mm to 400 mm form the active medium of TileCal [16]. Ionizing particles crossing the tiles induce the production of ultraviolet scintillation light in the base material (polystyrene) and this light is subsequently converted to visible light by wavelength-shifting fluors (the polystyrene is doped with 1.5% PTP as the primary fluor and with 0.044% POPOP as the secondary fluor). Over 460,000 scintillating tiles were produced for the Tile Calorimeter by injection molding

of individual tiles. The tolerance for all dimensions was held to 0.10 mm. Approximately 5% of the tile production was tested with a ^{90}Sr radioactive source and the results were used to characterize the light output of each small group of approximately twenty tiles in terms of maximum intensity and attenuation length. Two sources of raw polystyrene were used for tile fabrication; during assembly, the groups of tiles were sorted so that tiles with similar response were inserted in contiguous areas of the detector. Irradiation tests of tile/fiber assemblies indicated that in the first longitudinal sampling, for an integrated dose corresponding to ten years of operation at the LHC design luminosity, a light loss of less than 10% is expected. Smaller losses will occur in the other samplings, where the radiation dose is smaller.

Prior to insertion into the calorimeter, the tiles are inserted into a plastic sleeve, which both protects the tile and improves the scintillation light yield due to its high reflectivity of 95%. A mask pattern is printed on the sleeve to improve the optical uniformity. The resulting non-uniformity over the surface of a tile is generally below 5% for the sum of signals on both sides of the tile. Wavelength-shifting fibers placed in contact with the tile edges collect the scintillation light produced in the scintillators and convert it to a longer wavelength. Each fiber collects light from tiles located at one or two radial depths in the calorimeter and transmits it to the PMTs located inside the girder as shown in Figure 2.2. The fibers used have a diameter of 1 mm, are equipped with a double cladding and are characterized by an emission peak at 476 nm with a decay time of 6 ns. The fibers have an attenuation length of 325 cm at a wavelength of 430 nm, with a spread in attenuation length of 3% and in light output of 3%. To improve the light output, the fibers are aluminized at the end opposite to the PMT. The aluminium mirrors were deposited using magnetron sputtering on bundles of 1261 fibers.

The fibers are grouped together and coupled to the PMTs which are housed at the outer edge of each module. The fiber grouping is used to define a three-dimensional cell structure in such a way as to form three radial sampling depths, approximately 1.5, 4.1 and 1.8 λ thick at η=0. These cells have dimensions $\Delta\eta$ x $\Delta\phi$=0.1 x 0.1 in the first two layers and 0.2 x 0.1 in the last layer. The depth and η-segmentation of the barrel and extended barrel modules are shown in Figure 2.3. As the fibers coupled to each edge of the scintillating tiles are read out by two different PMTs this provides redundancy and sufficient information to partially equalize signals produced by particles entering the scintillating tiles at different impact positions.

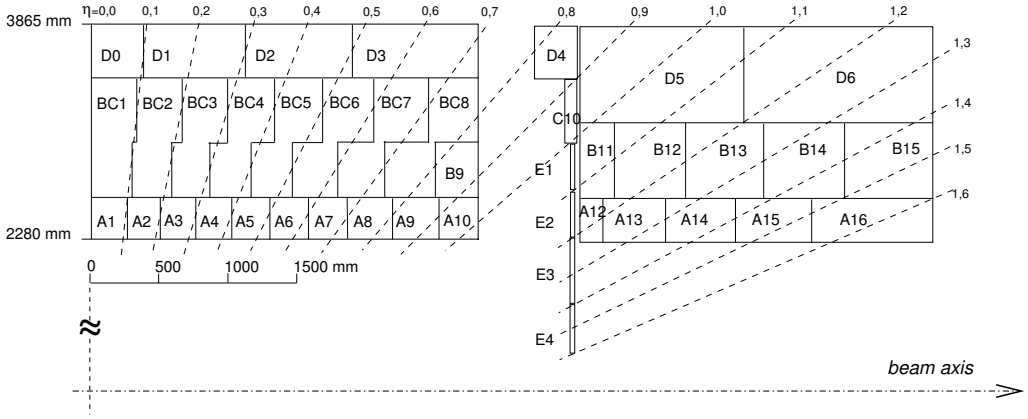


Figure 2.3: Segmentation in depth and η of the tile-calorimeter modules in the central (left) and extended (right) barrels. TileCal is symmetric about the interaction point at the origin.

2.3 Front-end Electronics

A block diagram of the tile-calorimeter front-end electronics [17] and readout components inside the drawer is shown in Figure 2.4. There are 256 superdrawers in TileCal, two for each Long Barrel (LB) module and one for each Extended Barrel (EB) module. The high voltages (HV) of the PMTs are regulated by special divider boards in the superdrawers.

The rest of the electronics is located outside the superdrawers, in the ATLAS electronics room: the HV power supply, the Level 1 Trigger, the RODs and the control electronics for the calibration systems.

2.3.1 Photomultiplier Block

A key element in the readout chain is the photomultiplier block. It is a mechanical structure comprising a steel cylinder and mu-metal shield for magnetic shielding, which contains a light mixer, a PMT, a voltage divider and the so-called 3-in-1 card (Figure 2.5). The light mixer is an optical plastic insert which mixes the light from the readout fibers to ensure uniform illumination of the photo-cathode. The PMTs with their compact 8-dynode structure are used to measure the scintillation light. All PMTs were burned in and tested for linearity, stability, dark current and operating voltage for a nominal gain of 10^5. The average operating

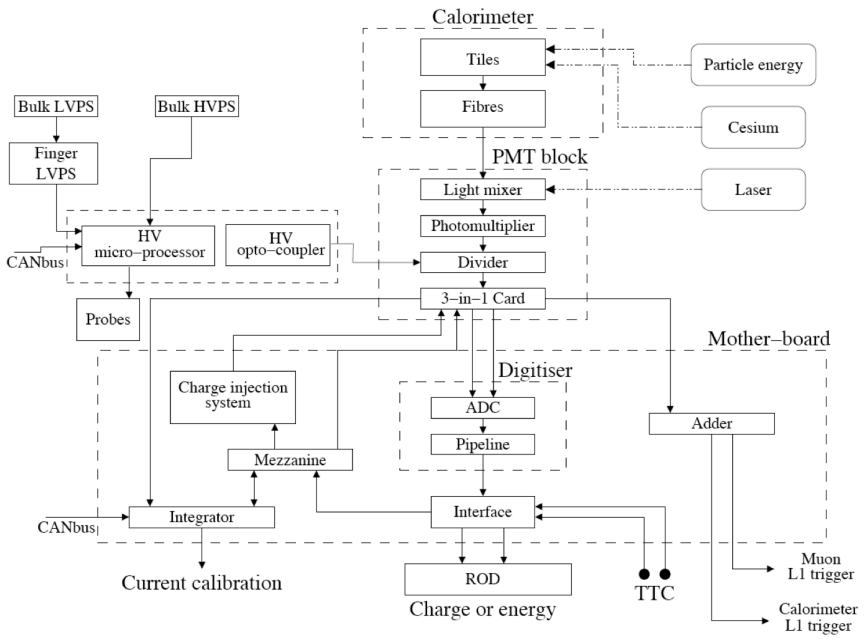


Figure 2.4: Block diagram of the TileCal front-end electronics.

voltage for nominal gain is 800 V. The assembled PMT blocks are inserted into precision slots inside the aluminium structure of the drawers, which ensure accurate placement of the light mixer relative to the fiber bundle for each readout cell. There is one PMT block assigned to each of the about 10000 fiber bundles in TileCal. The main components of a PMT block are the following (Figure 2.5):

- Photomultipliers. This device is responsible for converting the light signal from the fiber bundles into an electric charge. This photomultiplier should be able to work linearly in a wide range, from very low signals up to the signals coming from very energetic jets. After several studies it was found that Hamamatsu R5900 was the best choice. They are very compact (28 x 28 x 28 mm^3) and their dynode structure incorporates 8 amplification stages. The rise time is around 1.4 ns, which provides a fast response to the excitation. The needed voltage is around 800 V and the sensitivity to magnetic fields is very low, as well as the non-linearity, which is less than 1 %.

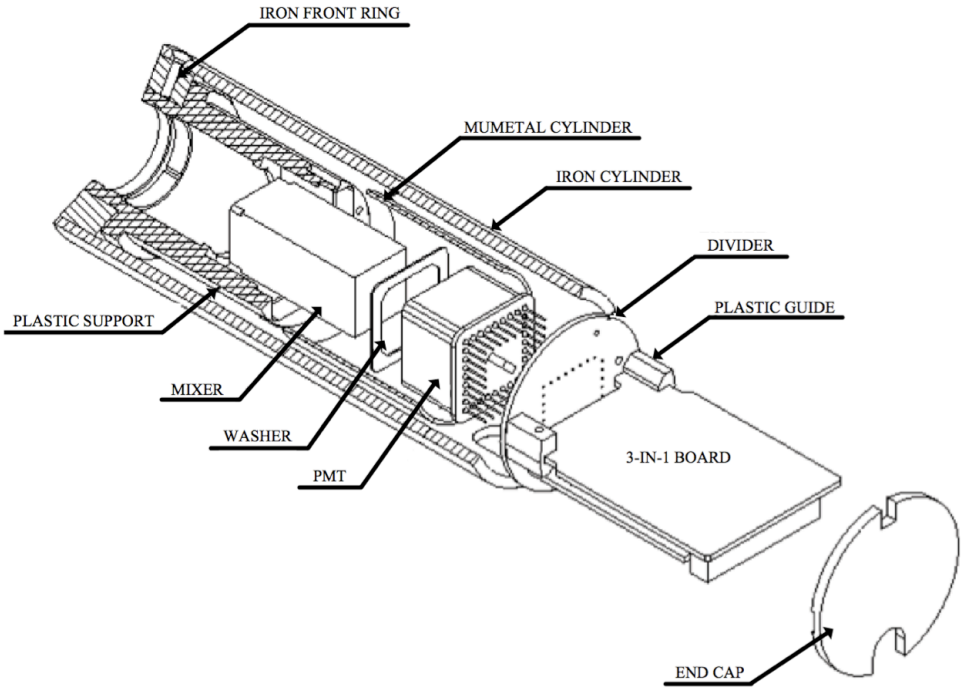


Figure 2.5: Sketch of the PMT block.

- The Light Mixers. Since the photomultiplier response depends over the photocathode surface position illuminated, a light mixer is used for mixing the light coming from all the fibers in the bundle, so that there is no correlation between the position of the fiber and the area of the photocathode receiving the light.

- Magnetic Shielding. The mu-metal and iron magnetic shielding in the PMT must prevent residual fields from the ATLAS solenoid and toroids from producing gain variations. It should provide a protection up to 500 Gauss magnetic fields in any direction.

- HV Dividers. The primary purpose of the divider is to partition the high voltage between the dynodes of the PMT. The TileCal divider also serves as a socket to allow the connection of the PMT to the front-end electronics without any interconnecting wires. This design minimizes the capacitance between the PMT and the electronics, which is important to reduce noise and unreliable connections.

- 3-in-1. The main functions of this board are to provide a high and a low gain shaped pulse for the digitizer boards, the charge injection calibration system and slow integration of the PMT signals for monitoring and calibration.

The TileCal Valencia-IFIC group tested a total of 1750 photomultiplier tubes of the TileCal.

2.3.2 Digitizer System

The PMT signals are shaped and amplified by the 3-in-1 cards inside the PMT block. Two output signals, high and low gain with a gain ratio of 64, provide the input for the digitizer system. The incoming pulses are digitized every 25 ns by 10-bit Analog-to-Digital Converters (ADC), using a sample clock that can be adjusted collectively for all ADC in steps of 106 ps. Adjustment is necessary to ensure that central sample is taken near the maximum of the pulse. The data is temporarily stored in pipeline memories awaiting a L1A. The pipeline latency is programmable up to 6.375 μs (2.5 μs being the minimal requirement). At each L1A a time frame containing up to 16 samples (7 during LHC operation) is transferred to a derandomizer buffer for its transmission to the ROD system. Headers and Cyclic Redundancy Check (CRC) trailers are added to the data for read out. In the standard readout mode, only one of the gains is read out, to reduce bandwidth. The high gain is read out by default unless it overflows or underflows. The analog part of the board contains components providing a voltage reference to the ADCs. It also contains two 8-bit Digital-to-Analog Converters (DAC) that provide a pedestal for the AC-coupled inputs. The digital part of the board contains two Tile Data Management Unit (DMU) Application-Specific Integrated Circuit (ASIC), the TTC receiver and decoder chip (TTCrx) [18], some Low-Voltage Differential Signal (LVDS) driver and receiver chips and a resistor field for hard-wiring the board address. Each TileDMU ASIC serves three channels. Therefore, inside a typical barrel superdrawer a row of 8 digitizer boards provide the necessary functionality (Figure 2.6) whereas 6 boards are sufficient for an extended barrel.

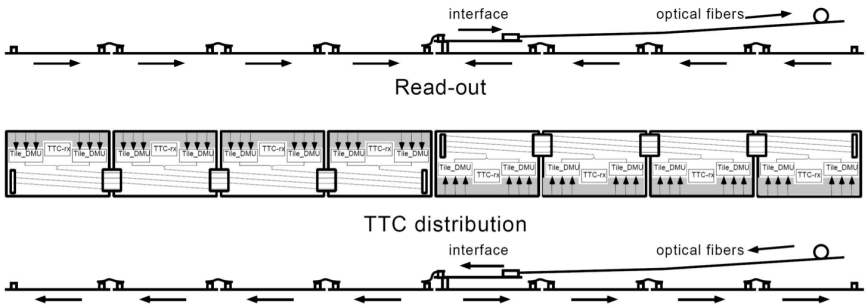


Figure 2.6: Diagram of the digitizer system in a superdrawer. Each Tile DMU ASIC processes the data from three channels whereas the TTCrx ASIC receives TTC signals. The part of the digitizer boards is analog.

2.3.3 Interface Board

There is one interface board per superdrawer [19]. It receives and distributes the TTC signals [20], collects and formats data from the digitizer boards, and transmits the digitized data via an optical link to the back-end system (Figure 2.6). The entire readout part of the interface has been duplicated for redundancy with two dedicated output fibers to reduce single event upset errors. The interface board also performs cyclic redundancy checks on the input and output data streams which are used in the back-end electronics to process data not affected by single upset errors and to perform online data quality checks.

2.3.4 Trigger Board

Special trigger summation cards, so called adder boards, are mounted on the mother-boards of the superdrawer. Each adder board receives the analogue trigger outputs from up to six 3-in-1 cards that form trigger towers, and perform an analogue sum of the signals and send two output signals via long cables to the Level 1 trigger system. One of these signals (tower signal) comprises the sum of all cells, whilst the second one (muon output) contains only the last layer cell. It also provides the sum of the signal in all four of the gap and crack scintillators. The muon output from these adder boards provides the signal from the scintillator covering the region $1.2<|\eta|<1.4$. Sixteen of these channels were used to trigger from the minimum-bias trigger scintillators during initial data-taking.

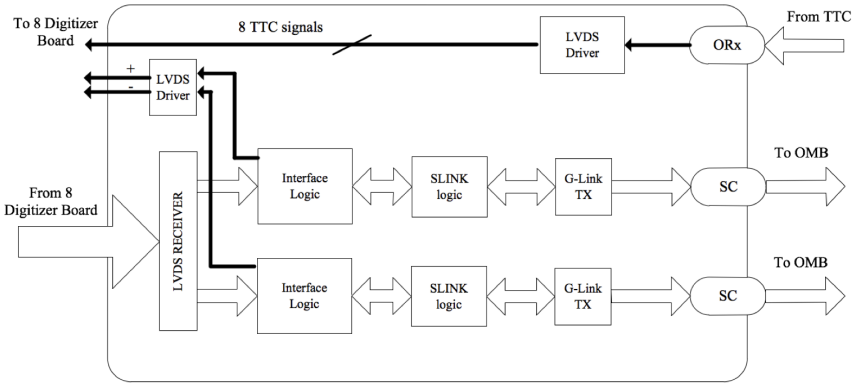


Figure 2.7: Diagram of the data flow and main functional blocks of the Interface Board.

2.4 Back-end Electronics Architecture

The back-end electronics is organized in four partitions, two servicing the readout of the Long Barrel and two servicing the readout of the Extended Barrels. Each back-end partition is equipped with its own TTC and ROD units. These units are physically split into a 6U Versa Module Eurocard (VME) ancillary crate, so called TTC crate, and a 9U VME and readout crate, so called ROD crate.

Figure 2.8 shows a picture of the TileCal back-end electronics racks. Left rack contains the TTC and ROD crates for side A of the detector, LBA (top) and EBA (bottom), the rack on the right contains the TTC and ROD crates for side C of the detector, EBC (top) and LBC (bottom). The center rack hosts the TTC Optical Couplers (TTCoc) which split the TTC optical signals from 1 to 32 links [20]. It also houses the patch panel for the front-end optical links, all TTC and readout links from the detector are connected to the back of the patch panel. Short optical fibers are routed from the front of the patch panel to the to the back-end electronics.

ATLAS standard VME modules are placed in the ancillary crate, along with specific modules for the TileCal. VME modules that are common to all calorimeters are placed in the readout crate. The modules in the crate are controlled by a Single Board Computer (SBC) 6U

Figure 2.8: Picture of the TileCal back-end electronic crates at USA15.

VME module. This SBC is a VP-110 model from Concurrent Technologies which is standard for all ATLAS. Each of the VME modules is briefly described in the following paragraphs.

2.4.1 Trigger Timing and Control Modules

Local Trigger Processor

The ATLAS Local Trigger Processor (LTP) receives timing and trigger signals from the CTP and injects them into the TTC system of the sub-detector [21]. The LTP allows also stand-

alone running by using local timing and trigger signal sources or by internal signal generation. The LTP can pass on signals to a sub-sequent LTP in a daisy-chained manner.

Local Trigger Processor Interface

The ATLAS Local Trigger Processor Interface (LTPI) is a VME module that interfaces multiple LTP modules with the CTP. The LTPI allows communication with other sub-detectors [22].

Trigger Timing and Control VME Bus Interface

The ATLAS TTC VME bus Interface (TTCvi) is a VME module that interfaces the local TTC system to the global TTC system. It delivers A and B Channel signals to the TTC transmitters for multiplexing, encoding, optical conversion and distribution to the TTCrx chips associated with the front-end electronics controllers [20]. The TTC channel A is used to transmit the Level 1 Accept (L1A) signal. The TTCvi incorporates a programmable L1A source selector and an internal trigger emulator for test purposes. The TTC channel B is used to transmit framed and formatted commands and data, which can be synchronous or asynchronous with respect to the LHC clock.

Trigger Timing and Control Emitter

The ATLAS TTC Emitter (TTCex) module is a laser transmitter module. It converts the TTCvi commands into optical TTC signals that arrive to the front-end electronics. It provides 10 optical outputs at a level $\sim$0 dBm.

Trigger Timing and Control Optical Coupler

The optical outputs of the TTCex are fanned out by a 1:32 TTCoc to broadcast to a total of 320 destinations. The module contains 2 biphase mark encoders driven by a common internal Voltage Controlled Oscillator/Phase-Locked Loop (VCXO/PLL) with very low jitter.

ROD Busy Module

The ATLAS ROD Busy module monitors the busy, measures it in bunch crossing units, and produces the sum of each of its 16 busy input lines, which can be conveniently masked. The ROD Busy module generates a VME interrupt after a programmed time-out. In the TileCal specifications, it receives the busy signal from the Trigger and Busy Module and the TTC Pluggable PCI Mezzanine Card (PMC) receiver (TTCpr) card. The busy output is sent to the LTP busy input.

Shaft Module

The Shaft module is a 6U VME module which controls the different calibration trigger requests foreseen in the TileCal. It is a specific VME board to share the calibration requests during physics runs. Each calibration request can be enabled and its firing timed with respect to the TTC signal (orbit) that clocks the turn of the LHC beam.

Trigger Timing and Control Receiver in PMC Form Factor

The TTCpr is a TileCal specific card. It is plugged into the Single Board Computer (SBC) of the TTC crate, which is used to make available the TTC information in the TDAQ framework for calibration runs. It was designed to provide Event ID, bunch crossing ID, and Trigger Type for each event in the data records. It provides a busy signal which is connected to the ROD Busy module.

Laser ReadOut Driver

The Laser ReadOut Driver is a 6U VME module that provides information from the Laser calibration system into the data-flow, and furnishes TTC signals to the Laser system. It is equipped with a High-speed Optical Link (HOLA) [23] card which provides data fragments to a ROB of the ROS of LBC partition through a ROL [24].

2.4.2 ReadOut Modules

Trigger and Busy Module

The Trigger and Busy Module (TBM) [25] is a 9U VME module for the ROD crate. It receives TTC signals from a optical link from the local TTC system. The signal is propagated to each ROD module through the P3/J3 connector, using the CP3 plane in the ROD crate. The TBM also gathers the busy signals through the CP3 from the eight RODs in the crate, and provides a busy signal to the ROD Busy module.

ReadOut Driver

The ReadOut Driver (ROD) is a 9U VME module that receives data from the superdrawers, through eight front-end links, each corresponding to one superdrawer (Figure 2.9).

It is equipped with two Processing Units which compute the energy and time of each readout channel, and it is coupled to a Transition Module (TM) [26], placed behind the ROD, which receives data fragments through the P2 and P3 connectors of the VME crate. The TM is equipped with two HOLA cards which transmit data fragments to the ROS through ROLs.

Eight RODs are installed in the ROD crate. Front and rear views of the ROD crate are shown in Figure 2.10.

The next Chapter includes a detailed description of the ROD motherboard, production and qualification tests of the modules, commissioning and operation of the ROD system.

Optical Multiplexer Board 9U

Radiation tests proved that radiation phenomena might cause malfunctions inside front-end electronics, as well as bit and burst errors in the data [27] . To reduce data loss due to radiation effect the TileCal collaboration decided to include data redundancy in the output links of the front-end by means of doubling the number of optical fibers per channel which would transmit the same data out of the detector. In this way if a radiation error occurs on data on one fiber, it is very probable that the other fiber be safe due to radiation space distribution properties inside the detector. Unfortunately, the ROD card has only one input connector for each front-end channel because the original design responds to initial specifications which did not

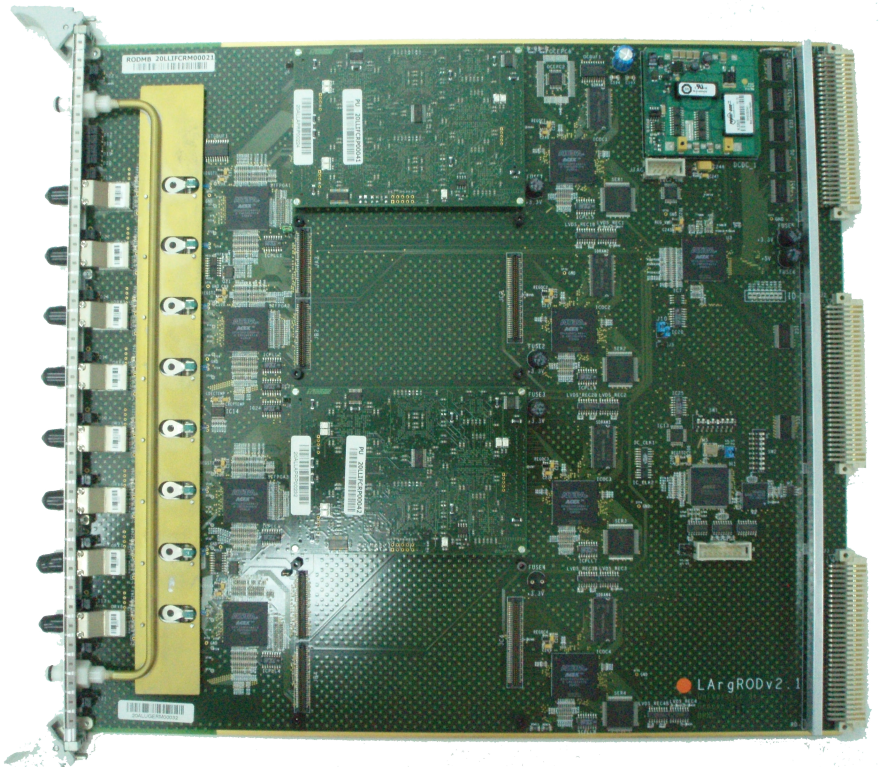
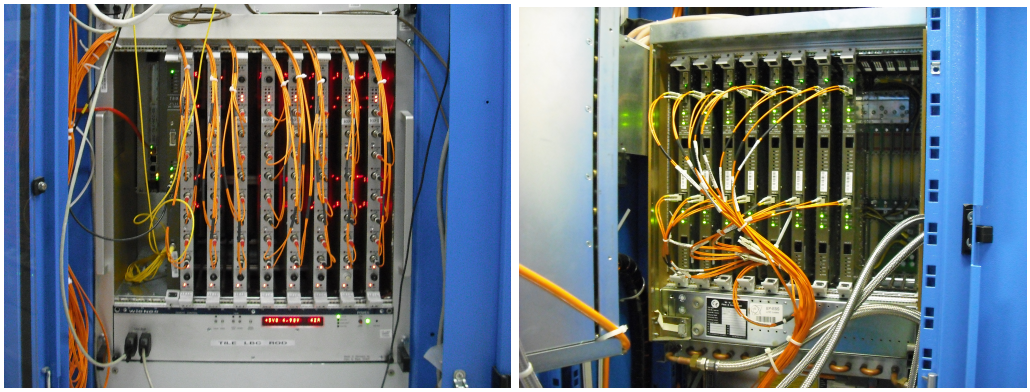


Figure 2.9: Picture of a ROD module equipped with two Processing Units.

consider radiation problems. For this reason and to take advantage of data redundancy and keep the original ROD design, an Optical Multiplexer Board (OMB) also known as Pre-ROD was envisaged. This board would be able to provide, in case of error in one link, the correct data to the ROD input by analyzing the CRC of the data packets on both fibers coming from the front-end. The OMB has been designed and produced by the TileCal Valencia-IFIC group. This 9U VME module will select data from one of the two front-end links and transfer it to the ROD. It can also be used to inject data to the ROD for test purposes.

A complete description of the board design, firmware implementation, production and qualification tests can be found in Chapter 4.



(a) (b)

Figure 2.10: Pictures of the front (a) and rear (b) views of the ROD crate.

2.5 TileCal ReadOut Principle

Particles produced in proton-proton interactions that travel through the TileCal deposit energy in the scintillating tiles which produces light. Wavelength shifting fibers turn the light into the visible blue which is then guided to photomultiplier tubes. The photomultiplier converts the light into an electrical analog pulse which is shaped and amplified in the 3-in-1 cards. The signals are summed in groups of five and the result is sent to the L1Calo. In parallel, the analog pulses are transmitted to the digitizers where the signals are digitized every 25 ns and the produced digital samples stored in the front-end pipeline memories, the so called Tile DMU. The CTP of the Level 1 Trigger sends a request signal to the front-end electronics for the events selected at a mean rate of 100 kHz. The request, in the form of TTC signals, consist of a Level 1 Accept (L1A) and a Bunch Crossing Identification (BCID). The back-end electronics receives these TTC signals from the CTP via the LTP and distributes them to the front-end through optical links. The TTC signals are distributed inside the superdrawer to the DMUs as shown in Figure 2.6. The samples corresponding to the requested BCID are transmitted to the Interface Board every time the DMU receives a L1A. The Interface Board builds up a data fragment containing the samples from all channels (PMTs) in the superdrawer and transmits it through optical links to the back-end electronics. The data flow rate is dynamically controlled using a busy feedback signal from the back-end electronics

to the CTP. The busy signal, generated by the readout modules when the input buffers are full, is combined together in the ROD Busy module and transmitted through the LTP to the CTP. This introduces dead time in which the CTP cannot request new events.

2.6 Calibration Systems

In general the readout chain of the signal can be separated intro three stages. Each stage can change its behavior with time and therefore has to be monitored and in some cases corrected (Figure 2.11). The three stages and possible effects are:

- Production of light in the scintillating tiles and propagation through the WLS fibers. Radiation damage and aging of fibers and scintillators can change their light yield. Such changes are typically slow and can be monitored by inducing light of know characteristics with the help of a radioactive source. In the TileCal a ^{137}Cs is used to calibrate the response of the scintillator tiles.

- Conversion of the light into a current at the PMT. Due to gain drift and deterioration of gain and quantum efficiency, the photomultiplier response may change. The injecting of a well known light pulse at this stage can be used to monitor such processes. A Laser system is used to induce a light pulse to the PMTs to monitor their response.

- Readout and processing of the calorimeter signal. Aging of electronic components can result in non-linearity. This can be monitored by injecting a defined charge into the electronics and comparing it with the signal given at the end of the readout chain. The 3-in-1 cards include a Charge Injection System to inject a programmable charge into the shaping and amplification circuit.

Therefore the TileCal inter-callibration and monitoring strategy is based on several calibration systems. The final channel reconstructed energy calibrated to GeV units can be expressed as:

$$E[GeV] = A[ADC] \times K_{ADC \to pC} \times K_{laser} \times K_{Cs} \times K_{pC \to GeV} \tag{2.1}$$

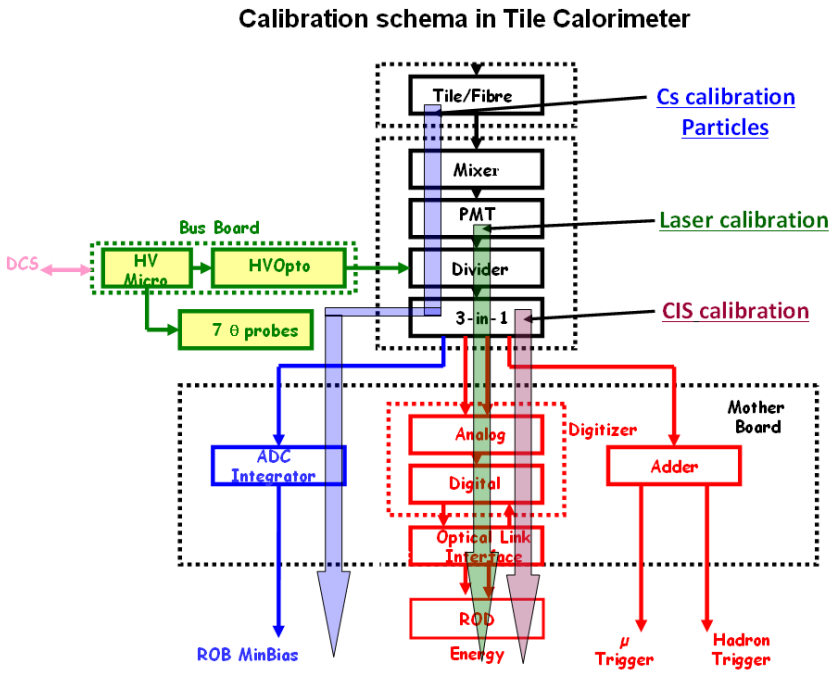


Figure 2.11: Diagram of the TileCal calibration systems.

where A[ADC] is the reconstructed amplitude of the pulse in ADC-counts. The term $K_{pC \to GeV}$ represents the conversion factor from charge (pC) to the electromagnetic scale energy (GeV) which was measured during the 2001-2003 test beam using electrons and muons. The other three constants in Equation (2.1) are periodically monitored and determined with the TileCal calibration systems. They provide corrections to the response of the scintillator and fibers, the PMTs and the electronics.

The ^{137}Cs Calibration System A radioactive source of ^{137}Cs is circulated through a system of steel tubes crossing each cell in Tile. This calibration is done every month with dedicated runs for about 6 hours when no beam is circulating in the accelerator. The cesium source induces light with the emittance of 662 keV γ-rays with an activity of around 330 MBq. The integrator circuit in the 3-in-1 cards measures the current from the PMT which is normalized to the cell size. The cesium system is the only calibration system that allows

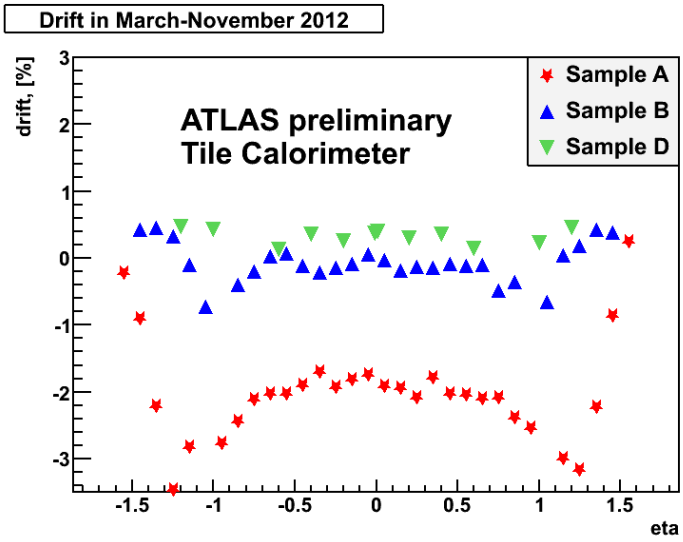


Figure 2.12: Average Drift of the PMT response as a function of the pseudo-rapidity for the three cell layers using the cesium calibration system during 2012.

to determine the response of the entire readout chain. This system is used to monitor the response of the optical elements, to measure the response of each cell and also to equalize the response of the calorimeter cells at the electromagnetic scale. The cesium calibration system determines the response of the optical chain with a precision better than 0.3%. Figure 2.12 shows the maximal signal change measured with the cesium calibration system during the 2012 data taking period as a function of the pseudo-rapidity in three longitudinal cell layers. Every point represents an average of 64 cells (over phi). The A cells, which are closer to the interaction point and therefore more irradiated, show in average the largest deviation over the period.

The Laser System The Laser system is used to monitor and calibrate the gain and linearity of the PMTs as well as the timing. The laser calibration is done twice per week using dedicated runs between beam fills and during physics runs in the LHC gaps dedicated to the calibration. The dedicated runs are used to equalize the response of the PMTs in time between two consecutive cesium runs whereas the laser pulses in empty bunches are used to

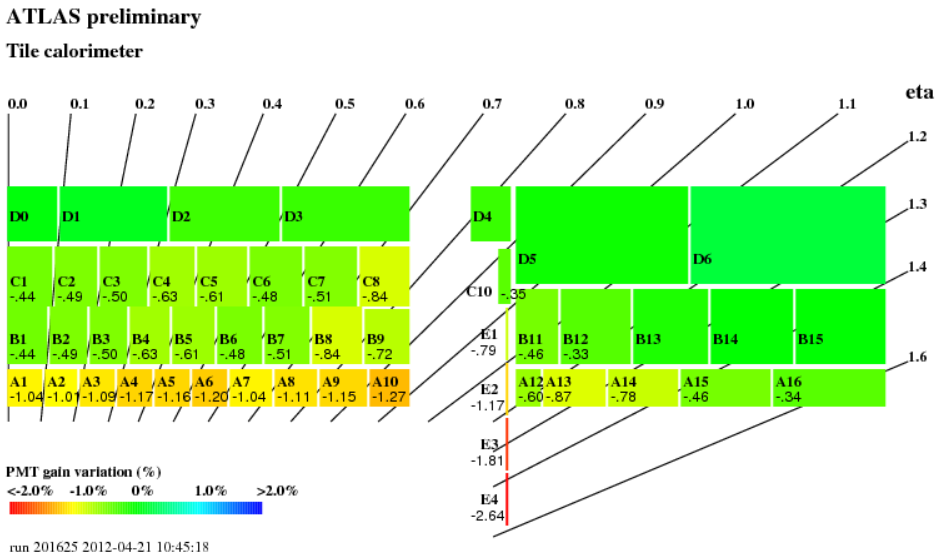


Figure 2.13: Mean PMT gain variation per cell type measured with the laser system between two cesium runs in April 2012.

monitor the stability of the timing. The laser emits a pulse of light of 532 nm to each PMT. This wavelength corresponds to the typical wavelenght of the light produced by particles crossing the detector. The light produced is split and distributed to all the TileCal PMTs. Furthermore, the amplitude of the produced light pulses is monitored inside the Laser system using a set of photodiodes. Figure 2.13 shows the mean gain variation for each cell type where the gain in each PMT was measured using the laser system. For each cell, the gain variation is defined as the mean of the gaussian function that fits the gain variation distribution of the channels associated to this cell. The observed down-drift of $< 2\%$ mostly affects A cells which are closer to the interaction point and therefore experience a higher current in the PMTs.

The Charge Injection System The Charge Injection System (CIS) is designed to calibrate the relative response of the pulse readout electronics for all calorimeter PMTs and to monitor the variations with time. Each PMT channel has two analogue paths, the high and low gain

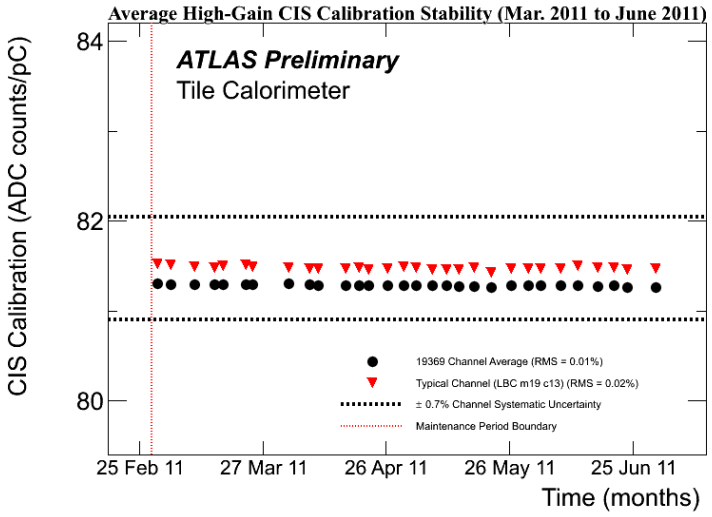


Figure 2.14: High gain CIS calibration stability during the 2012 data taking period.

(designed for a conversion factor of 82 counts/pC and 1.3 counts/pC, respectively), digitized by a 10-bit ADC, covering a range of 800 pC (corresponding to a energy deposition of about 700 GeV). Each channel is equipped with calibration capacitors, charged from a voltage source and discharged into the input electronics. The resultant waveform is similar to the one produced by the PMT for a given charge, except for an amplitude 10% larger and a Full Width at Half Maximum (FWHM) 10% smaller. Dedicated CIS runs are taken twice per week when the beam is absent, scanning the full range of charges for both gains, allowing for the determination of the ADC-counts/pC ratio for each channel. Figure 2.14 represents the evolution of the high gain CIS calibration constants for the 2012 data taking period. The stability of the calibration constant is within the $\pm 0.7\%$ channel systematic uncertainty in the whole period.

The Integrator System The Minimum Bias (MB) events in ATLAS are inelastic pp collisions with low momentum transfer, whose rate is proportional to the LHC luminosity. These events produce a non-negligible occupancy of the TileCal cells, with rates uniform in the azimuthal angle and moderately dependent on the pseudo-rapidity. Since the MB current,

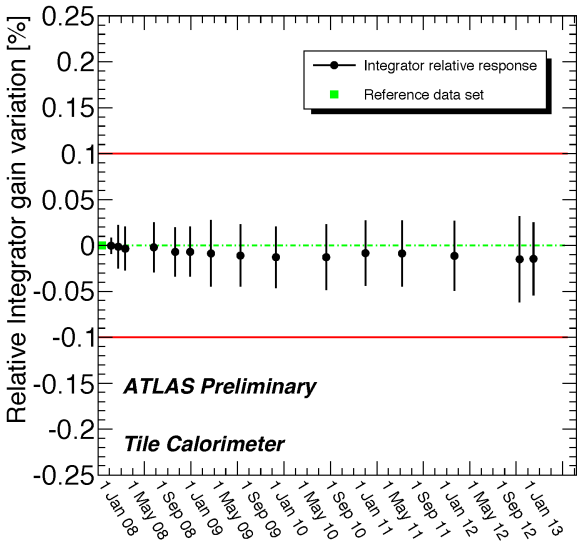


Figure 2.15: Relative variation of the integrator gain used by the Cesium calibration system as a function of time obtained by comparing the gains of all the channels of Tile.

averaged over milliseconds, is almost constant and proportional to the interaction rate, it can be used to monitor the calorimeter response, the relative luminosity and the beam quality during physics runs. Dead and hot channels are also identified in MB data. For each PMT channel the anode output is DC coupled to an integrator circuit based in an operational amplifier. This has a configurable time constant between 10-20 ms and a gain selectable from six predefined values (as the MB current changes with the position of the cell on the calorimeter, as well as with the luminosity). All integrator outputs, multiplexed to a 12-bit ADC, are readout every 2 ms through a dedicated Controller Area Network (CAN) bus. This integrator system is also used to readout the cesium calibration data. The stability of individual channels in the system is better than 0.05% whereas the average stability better then 0.01% as presented in Figure 2.15.

Chapter 3

TileCal ReadOut Driver System

3.1 Introduction

In the previous Chapter we introduced the ATLAS Tile Calorimeter front-end electronics, the basic principle of the detector based on scintillator cells, fibers and PMTs, how the events are selected by the first level of trigger and how the data is transmitted to the back-end electronics.

In the back-end the data are received by the ROD which is a sub-detector dependent implementation located in the ATLAS acquisition chain between the first and the second trigger levels. At the Level 1 Trigger rate, the ROD system processes in real time the data from 9856 front-end PMTs in real time, i.e. in less than 10 μs. The processed data are transmitted through ROLs to the ROS located in the Level 2 Trigger (Figure 3.1).

3.2 Structure of the Back-end System. The Regions of Interest

A RoI is a geometrical region of the ATLAS detector described in units of pseudo-rapidity (η) and azimutal angle (ϕ) defined by the Level 1 Trigger [28]. These are used by the Level 2 Trigger to perform HLT algorithms on the data within the ROI. TileCal is segmented into

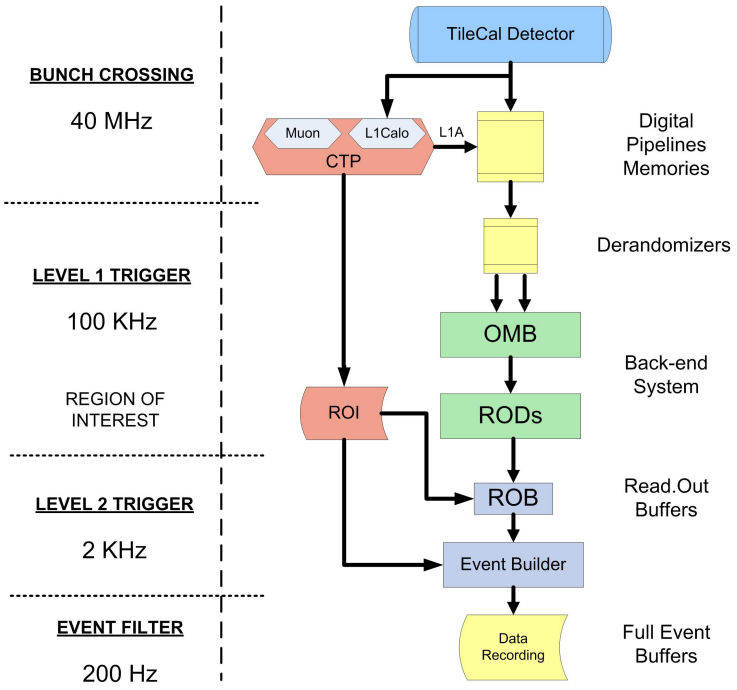


Figure 3.1: Trigger Levels for TileCal detector including the OMB modules.

different η regions with ϕ symmetry with the purpose to serve different physics searches in transverse momentum (p_T). TileCal electronics are divided into two high p_T regions (-0.7<η<0 and 0<η<0.7) and two low p_T regions (-1.2<η<-0.7 and 0.7<η<1.2). The two high p_T regions correspond to the Long Barrel of the detector whereas the two low p_T regions are the Extended Barrels. Each of the four detector partitions (two in the LB and two EB) is associated to a Trigger and DAQ (TDAQ) partition. Since each ROD processes the data from 8 consecutive front-end superdrawers, the output data of a ROD is associated with a fixed area of the detector, which optimizes the access of the HLT to the data of one single ROI.

Moreover, each of the four TDAQ partitions is associated to a back-end hardware system composed by a TTC and a ROD crate. This allows each partition to operate independently as a separate detector.

3.2.1 The TTC Crate

The TTC crate provides communication with the CTP for trigger management, reception of L1A trigger signals as well as the TTC commands and data to handle reset messages, calibration triggers and control and test parameters [22]. In addition, the TTC crate collects the busy signal from RODs and transmits them to the CTP when the ROD input buffers are nearly full. The individual modules included in the TTC crate were described in Chapter 2.

In combined operation mode, mainly used for physics data taking, the CTP receives the bunch crossing and the orbit signal from the LHC machine. The CTP then broadcasts to each ATLAS TTC partition the 40.08 MHz bunch crossing clock, Event Counter Reset (ECR), Bunch Crossing Reset (BCR), Trigger Type (TType), L1A, orbit and pre-Pulse signals. It also receives a logical sum ('OR') of the busy signals and 3-bits for calibration requests from each partition. Figure 3.2 shows the modules and connections of a TTC crate in the global operation mode.

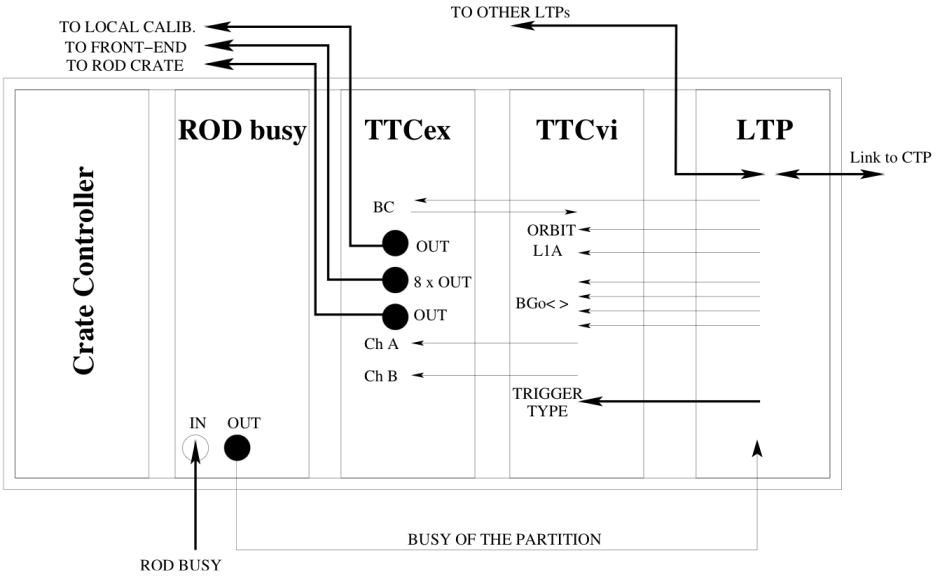


Figure 3.2: Sketch of the TTC crate with the TTC modules and connections in Global operation mode.

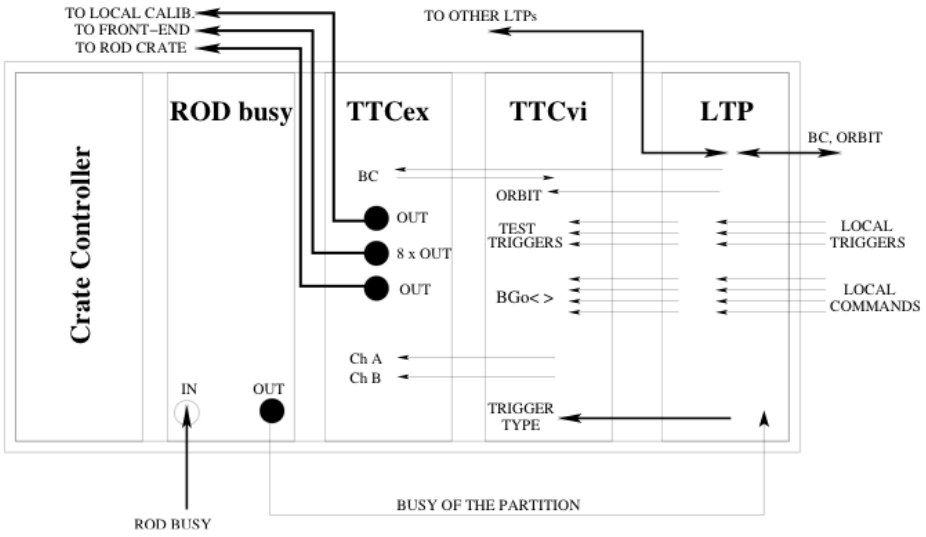


Figure 3.3: Sketch of the TTC crate with the TTC modules and connections in Local operation mode

In the local mode, all signals can be generated internally by the LTP or the TTCvi. Figure 3.3 shows the connections inside the TTC crate during local mode run. The local mode is used to operate the different ATLAS sub-detectors in standalone mode, which is used mainly used for local calibration. Moreover, in the local mode it is possible to operate each of the four TileCal TTC partitions independently.

3.2.2 The ROD Crate

Each of the four ROD crates contains a crate controller, a TBM and eight ROD modules with the eight associated Transition Modules. The system is prepared to operate eight additional OMB in each crate although currently they are not installed (Figure 3.4). As in the TTC crate the controller is a VP-110 SBC from Concurrent Technologies which provides communication with the modules through the VME bus. The TBM is the interface between the ROD and TTC crates. The TBM is used to collect the busy status of all the RODs in the crate through the backplane and to transmit the result to the TTC crate and to receive and distribute the TTC information to all the ROD modules. The eight ROD modules are used for data

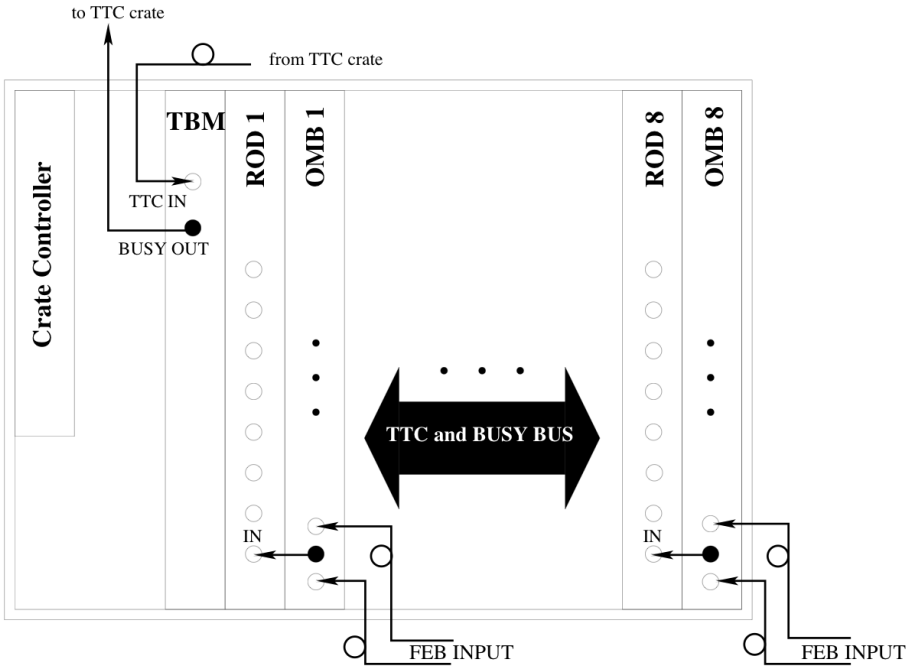


Figure 3.4: Sketch of the ROD crate.

processing and are linked to the ROS through its associated TMs placed in the rear part of the crate. Each ROD receives and process the data of one eighth of one of each partition. According to the ROIs found in the event by the CTP, the HLT processors receive a sub-set of the event data from the ROS, which reduce considerably the network bandwidth required for the trigger decision.

3.3 Trigger and Busy Module

The TBM is the interface between the TTC and ROD crates. The TBM receives the TTC information through an optical fiber from the TTC system.Then, the TTC signals are transmitted to the all the RODs placed in the crate through point-to-point transmission lines on the custom P3 VME backplane. On the other hand the TMB implements a logical OR of the RODs busy signals received through the same VME backplane. The result is transmitted to

the TTC crate and finally to the CTP to hold the generation of Level 1 Trigger acceptance signals in case of any ROD cannot accept more data because of full input buffers.

3.4 ROD Module

The ROD module is based on a standard 9U VME64x board and is equipped with two Processing Unit (PU) which are the Digital Signal Processor (DSP) based pluggable daughterboards responsible for the data processing. The block diagram of the ROD and its associated TM is shown in Figure 3.5. The TM provides full duplex communication between the ROD and ROS which is used to transfer data to the ROS and to receive the ROS buffers status. The ROD must stop transmitting data and transmit the busy signal to the CTP in case the ROS buffers are full (XOFF).

The Optical Receiver There are eight custom Optical Receivers (ORx) placed in the front-panel of the ROD [29]. The data coming from one superdrawer is received in each ORx with an effective data rate of 640 Mbps. These ORx modules have an optical to electrical converter and an amplifier to provide constant-level output and controlled edge electrical signals to comply with Positive Emitter-Coupled Logic (PECL). These signals contain the serial data to be sent to the G-Link deserializer chipsets in the ROD motherboard.

The G-Link HDMP-1024 Chip The data received through the Optical Receivers are deserialized by G-Link chips (HDMP-1024) mounted in the ROD [30]. The ROD G-Link deserializes the data previously serialized at the other part of the optical link. Then, the deserialized data is transmitted to the Staging Field Programmable Gate Array (FPGA) at a data rate of 16bit@40 MHz together with some additional control signals used to detect data and control packets. Two local oscillators chips and clock buffers provide 40.00 MHz reference clocks for the internal PLL of the G-Link chips.

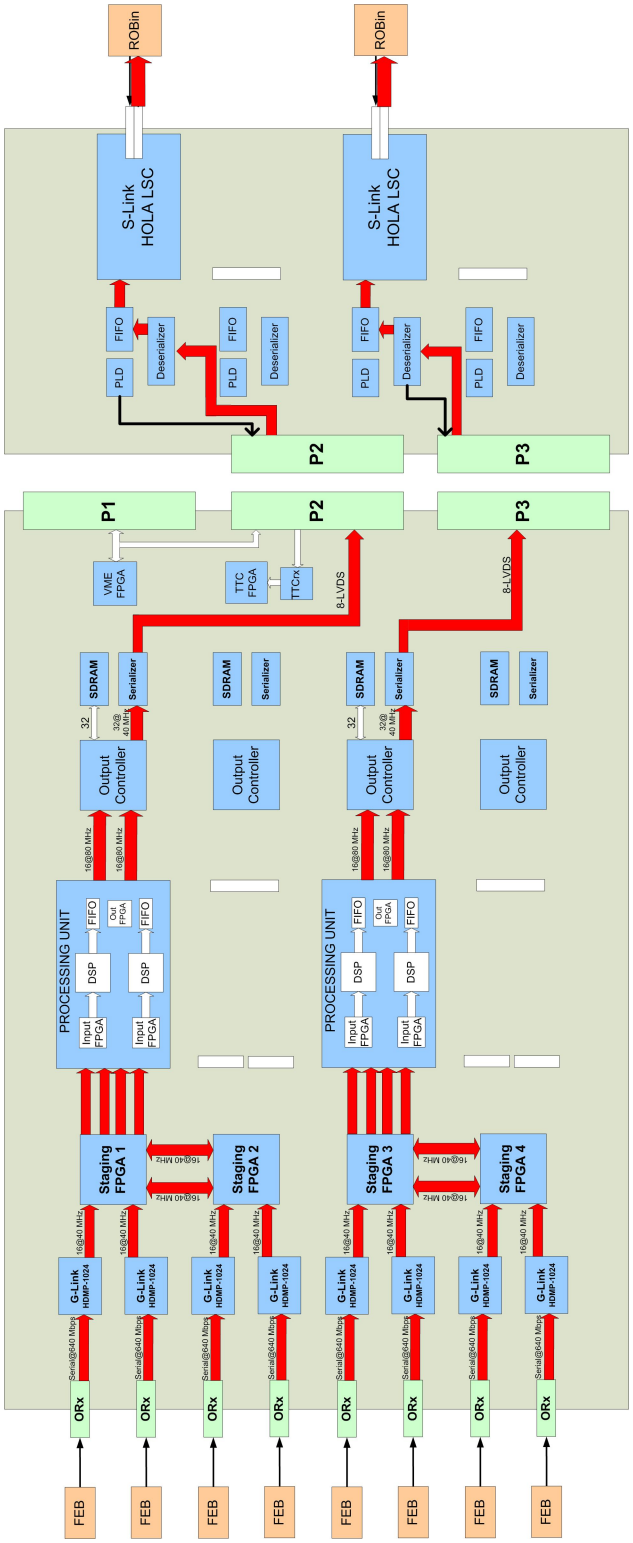


Figure 3.5: Block diagram with the main components of the ROD motherboard and the associated Transition Module. The red arrows show the data flow path.

The Staging FPGA The Staging FPGA constitutes the input data distributor of the ROD. There are four Staging FPGAs in the ROD each of them receiving the deserialized data directly from two G-Link chips. The Staging FPGAs can be configured to transmit the data received from the deserialized chips to the corresponding Processing Unit or to the neighboring Staging FPGA. In TileCal specifications the data from four ROD input links are gathered in one single Staging FPGA (two directly from the deserialized chips and two through the neighboring Staging FPGA) and then transmitted to one Processing Unit. The Staging FPGA is also responsible for controlling and monitoring the two G-Link chips. It provides communication with the VME backplane to configure, reset and to monitor the temperature of the G-Link chips. The G-Link temperature is very important in order to monitor the chips performance [31]. The temperature measurement is implemented with a thermistor glued to each G-Link which is sampled by an ADC that uses an internal reference voltage, an input analog multiplexer and a serial digital output to the Staging FPGA. The Staging FPGAs provide a clock de-skew between the G-Link clock (40.00 MHz) and the Processing Unit clock (40.08 MHz) with a dual-clock First-In First-Out (FIFO) memory implemented inside the device.

The Output Controller The Output Controller (OC) FPGAs are the output data distributors of the ROD. There are four OC mounted in the ROD but only the ones adjacent to PU are used in TileCal specifications. Each OC reads out the data from one PU and builds a so-called ROD data fragment. This fragment, containing the processed data of four superdrawers, is transmited to the ROS through the HOLA card located in the TM [23]. This data path is also called a ROL. The OC provides full duplex communication with the ROS. If the XOFF signal is asserted by the ROS system indicating that its input buffers are nearly full, the OC will stop transmitting data to avoid memory corruption due to overwritting. The output bandwidth of a TileCal ROL is 1.28 Gbps and it determines the maximum fragment size that can be transmitted for a given Level 1 Trigger rate without introducing dead-time in the detector. For the nominal Level 1 Trigger rate of 100 kHz the maximum ROD fragment size is 400 32-bit words.

The ROD TTC system The TTC system of the ROD is composed by a TTCrx chip responsible for the decoding of TTC signals and a TTC FPGA to manage and broadcast the TTC information to the PUs [18]. The TTC LVDS electrical signals distributed by the TBM through the P3 VME backplane are received and decoded by the TTCrx ASIC chip at ROD level. All TTC connections between the TBM and the RODs within the crate are 100 Ω point-to-point (which corresponds to the characteristic impedance of the CP3) which are ended with a 100 Ω resistor near the ROD LVDS receiver. The TTC signals contain information about the A and B channels of the TTCvi as well as trigger information and commands. The decoded information is then transmitted to the TTC FPGA which distributes the TTC information to each PU in a serialized point to point connection. The information and signal managed by the TTC FPGA are:

- The TTC clock (40.08 MHz).
- The L1A trigger signal.
- The BCID (12-bits): sampled bunch crossing when an interesting event occurs (L1A). It is internally implemented as a counter incremented with the TTC clock and cleared by the BCR signal.
- The TType (8-bits). This value is used to identify the type of event being acquired and to apply the appropriate treatment and algorithm at the DSP level (e.g., Physics , Calibration, Zero Bias trigger, etc).
- The Event Identifier (EVID) (32-bits) : The 24 Least Significant Bits (LSB) contain the value of a counter which is incremented by the L1A signal and reset by the ECR signal. The 8 Most Significant Bits (MSB) corresponds to a counter incremented by ECR signal. The EVID uniquely identifies an event during a Lumi Block of a run.

Another important feature of the TTC FPGA is to provide the clock to all the devices in the ROD through dedicated Zero-Delay clock buffers. During normal operation mode the LHC clock received through the TTCrx chip is distributed throughout the ROD board. The TTC FPGA automatically switches to a local oscillator if it detects problems in the LHC clock reception. It is possible to manually select the local oscillator through the VME backplane.

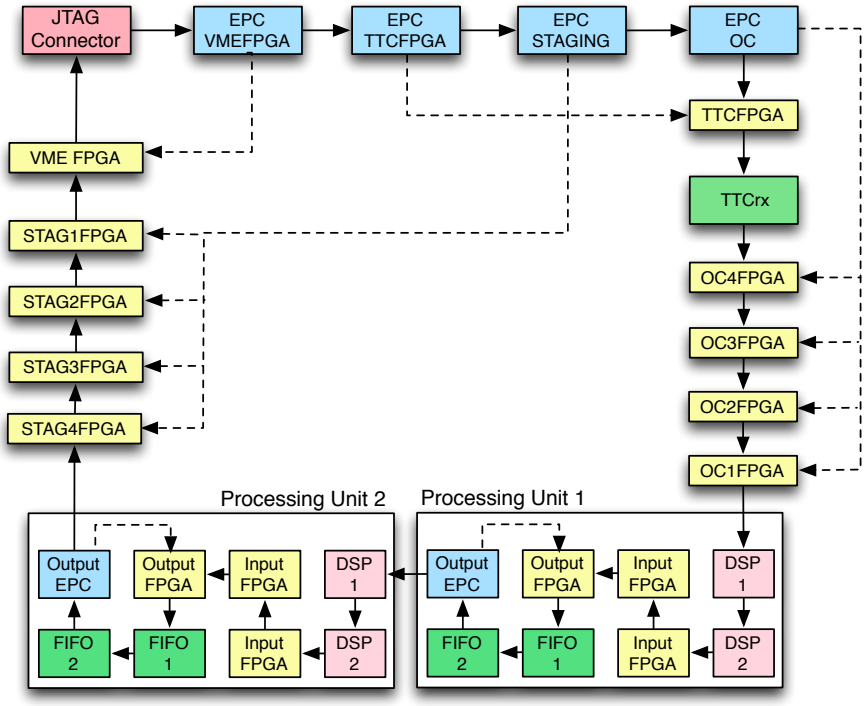


Figure 3.6: Connection of programmable devices within the JTAG chain. The solid black arrows show the transmission of JTAG signals whereas dashed black arrows show the configuration of FPGAs from EPC modules on every startup of a ROD module.

The ROD VME interface There is one VME FPGA in each ROD to provide communication between the ROD crate controller and the rest of the devices in the ROD board. On one side the communication with the ROD crate controller is based on a standard VME64x protocol that allows single read/write cycles as well as block data transfer. On the other side the communication with all the devices in the ROD board is implemented through a custom serial protocol. The protocol is based in 5 serial lines; one unidirectional line managed by the VME FPGA is used to indicate a data or and address word; 4 bidirectional lines are used to transmit the serialized 4 bytes of the 32-bit word. The communication with the the Staging FPGAs is implemented with a daisy-chained bus to avoid extra routing lines in the PCB.

This communication allows the configuration of the ROD devices, the code booting and monitoring of the PUs as well as remote access to the Joint Test Action Group (JTAG) chain

to update the firmware of the ROD programmable devices. Figure 3.6 shows the components within the JTAG chain. The JTAG connector is used to in-situ configuration of any FPGA or EPC memories in the chain. The VME FPGA provides remote access to the JTAG chain for configuring the EPC non-volatile memories. In order to transfer the configuration to the FPGAs after the reprogramming of the EPC memories a initialization of the board power is needed. Another important function of the VME FPGA is the management of the ROD Busy signals and interrupts handling. The VME FPGA receives the Busy status of each DSP and the logical 'OR' of these Busy signals is transmitted to the TBM through dedicated point-to-point connections in the P3 VME backplane. In addition it provides monitoring of the ROD Busy status during a configurable period of time. It is used to detect persistent Busy status from a given DSP which would indicate problems in the data-flow that would require external intervention.

3.4.1 Processing Unit

The TileCal ROD holds two PU mezzanine cards. Each PU has to process the data of 4 TileCal superdrawers. The PU is equipped with two Input FPGAs, two DSPs and two Output FIFOs. All these dual devices are used to get double processing power in a single PU and they are responsible for I/O data-flow and digital signal reconstruction. In addition an Output FPGA provides communication with the VME and TTC interface of the ROD motherboard as shown in Figure 3.7.

Input FPGA There are two Input FPGAs (Cyclone EP1C6F256C8 from Altera) in each ROD PU each of them receiving the input data from two superdrawers. Then, the data events are formatted to optimize the computation of the digital filtering algorithms in the DSP. The samples of each channel are packed in four consecutive 32-bit words (Figure 3.8). Moreover, the Input FPGA performs data integrity checks and the result is transmitted in dedicated data quality fragment to the DSP. For calibration runs, where the input data rate is considerably lower than for physics operation, the Input FPGA can be configured to transmit the raw data received from the superdrawers without any additional formatting. When the Input FPGA receives a complete event it sends an interrupt to the corresponding DSP and

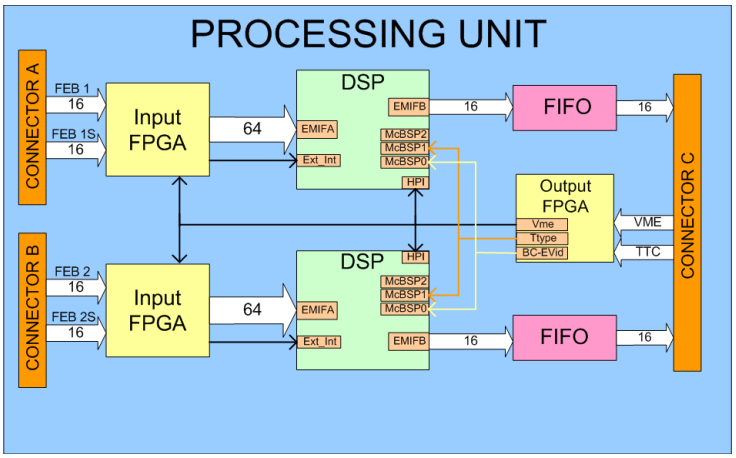


Figure 3.7: Block diagrams of the Processing Unit.

the whole event is transferred to the DSP input buffer. The data events with unexpected size are deleted by the Input FPGA to avoid the processing of corrupted data in the DSP.

Digital Signal Processors The ROD processing units are equipped with DSPs from Texas Instruments TMS320C6414^{TM} in order to execute reconstruction algorithms in real time [32]. The choice of DSPs over similar devices, e.g. FPGAs, was justified because they can be programmed in high level languages, such as C language, which are simpler to maintain and to adapt to the detector requirements. This is needed because the selected device should be flexible for future upgrades of the reconstruction algorithms depending on the LHC conditions which may vary along its life. Moreover, the commercial DSPs implement Multiply-Accumulative (MAC) instructions easily, which are the base of standard data filtering algorithms. They have also an internal local memory where input and output data is stored using a circular buffer. Another useful feature of the DSPs is the access through a host port interface and serial ports which allow to reconfigure the DSPs at run time, read histograms from these devices and load new calibration constants for the reconstruction algorithms.

The TMS320C6414^{TM} DSP belongs to the TMS320C6000^{TM} family of fixed-point DSPs from Texas Instruments. The TMS320 DSPs have an architecture specifically designed for real time processing. They use a high performance advanced VelociTI^{TM} Very Long Instruction

32-bit words			
	10 bits	10 bits	10 bits
	Sample1 CH3	Sample1 CH2	Sample1 CH1
	Sample2 CH3	Sample2 CH2	Sample2 CH1
	Sample3 CH3	Sample3 CH2	Sample3 CH1
	Sample4 CH3	Sample4 CH2	Sample4 CH1
	Sample5 CH3	Sample5 CH2	Sample5 CH1
	Sample6 CH3	Sample6 CH2	Sample6 CH1
	Sample7 CH3	Sample7 CH2	Sample7 CH1

32-bit words			
	10 bits		10 bits
	Sample1 CH1		BCID
	Sample3 CH1		Sample2 CH1
	Sample5 CH1		Sample4 CH1
	Sample7 CH1		Sample6 CH1
	Sample1 CH2		BCID
	Sample3 CH2		Sample2 CH2
	Sample5 CH2		Sample4 CH2
	Sample7 CH2		Sample6 CH2
	Sample1 CH3		BCID
	Sample3 CH3		Sample2 CH3
	Sample5 CH3		Sample4 CH3
	Sample7 CH3		Sample6 CH3

Figure 3.8: Organization of the data in the Input FPGA.

Words (VLIW) architecture (VelociTI.2^{TM}). This architecture consists on multiple execution units that can run in parallel which allows to perform multiple instruction in a single clock cycle.

Figure 3.9 shows the block diagram for the TMS320C64x. This DSP has separated program and data memories that can be used as program and data cache, respectively. It is also provided with peripherals such as a Direct Memory Access (DMA) controller, power-down logic, External Memory Interface (EMIF), Multi-Channel Buffered Serial Ports (McBSP) and parallel Host Ports Interface (HPI).

The DSP CPU is composed of a program fetch unit, instruction dispatch unit and instruction decode unit that can deliver up to eight 32-bit instructions to the functional units every CPU clock cycle. There are two data paths (A and B), each one provided with four functional units, three ALUs and one multiplier, and 64 32-bit general purpose registers. The internal memory of the chip is organized in a level-one cache (L1) of 32 kB, separated in 16 kB of program memory space (L1P) and 16 kB of data memory space, and a unified program and data memory space of 1024 kB that can be used as SRAM or level-two cache. The program memory space is direct-mapped, it has a 256-bit wide data path to the DSP core, so that the DSP core may fetch up to 8 32-bit instructions every single clock cycle. The data memory space is two-way associative, it allows two simultaneous data accesses from the two DSP core

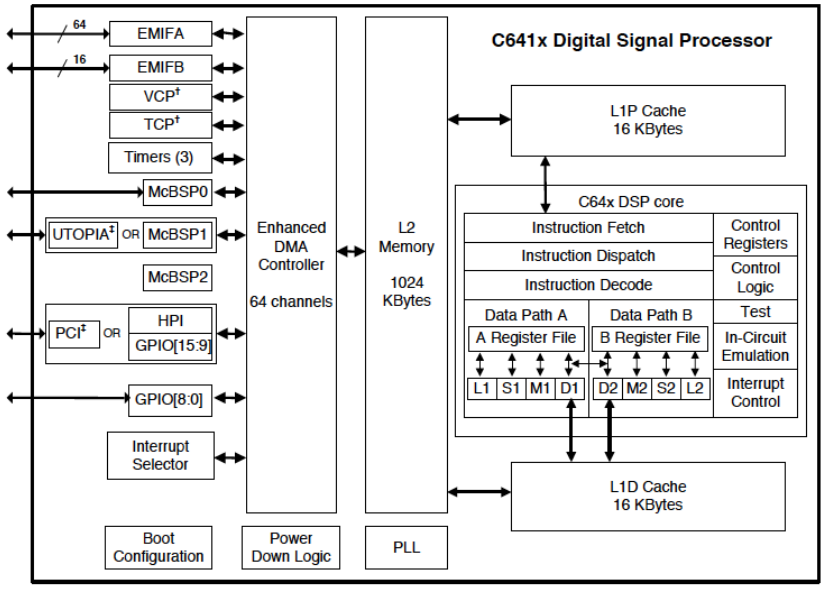


Figure 3.9: Diagram of the TMS320C641x DSP.

data paths, so the DSP core can load or store two 64-bit values in a single data cycle. DMA and EDMA controllers transfer data between address ranges in the memory map without intervention by the CPU, and a host processor can directly access the memory space through a HPI port. The EMIF bus acts as host port and as I/O port that can operate in synchronous or asynchronous mode. The McBSPs are serial ports. These ports can buffer serial samples in memory automatically with the aid of the DMA controller. The DSPs are also provided with timers and power-down logic. Table 3.1 shows a summary of the main features of the TMS320C6414 DSP.

Output FIFO The Output FIFO is an IDT72V253L75BC 3.3 V high density super synchronous 18-bit bus FIFO organized in 4k 18-bits words. The DSP controls the data transmission to the Output FIFO through the programmable almost full and empty flags. On the other side, the Output Controller of the ROD is continuously checking the status of the Output FIFOs of the PU to retrieve any available data.

Features	**TMS320C6414**
Cycle time	1.39 ns
Clock rate	720 MHz
Instruction/cycle	8 32-bit instructions
Functional units	6 ALUs, 2 multipliers
Data memory (L1D)	16 kB
Program memory (L1P)	16 kB
Level-two cache (L2)	1024 kB
EDMA	64 channels
EMIF	1x16-bit, 1x64-bit
Host port	32-/16-bit
McBSP	3
Timers	3x32-bit
Core supply	1.4 V
IO supply	3.3 V

Table 3.1: Main features of the TMS320C6414 DSP.

Output FPGA The Output FPGA implements the interface between the PU components and the ROD motherboard providing the TTC information and the communication with the VME bus. The TTC information is received in the Output FPGA from the TTC FPGA through two point-to-point serial connections and it is transmitted to the DSPs via 2 serial ports. One serial line is used to transmit the Trigger Type which is received in the DSP through the McBSP1 port. On the other hand, the BCID and EventID are received in the DSP through the McBSP0 port. The Output FPGA is the PU interface with the ROD VME layer. This interface is used to boot and initialize the DSP code and to retrieve monitoring information from the DSP internal memory through the 16-bit HPI. It also provide full duplex communication via the McBSP2 with each DSP which is used to transmit commands and configure the DSPs. Moreover, the Output FPGA is used to boot the code and to transmit the configuration to the Input FPGAs through dedicated serial lines.

Transition Module The TM is a standard 9U VME rear transition module that is placed in the rear part of the VME crate and coupled to each ROD board. The output data of each ROD PU is transmitted to one HOLA plugged in the TM. The data is transmitted between the ROD and the TM through dedicated point-to-point connection in the P2 and

P3 connectors of the VME backplane. Due to the limited number of pins in the P2 and in the custom P3 backplanes, serializers and deserializers are used in the ROD and in the TM respectively.

ROD data-flow paths There are three main important paths to be emphasized due to the different tasks they perform in the board (Figure 3.5).

- Data-flow: the input data coming from one of the eight optical fiber is received by eight optical receivers and deserializer G-Link chips to be distributed into one of the four Staging FPGAs. The main function of the Staging FPGA is to distribute the superdrawer data into the PUs. In the PU, the data is processed and formatted in less than 10 μs and the output data are buffered in the PU FIFOs and received by the OC FPGA. The OC sends the event fragment to the ROS through the TM HOLA cards. The data-flow path is shown in Figure 3.5.

- TTC and Busy signals: the TBM transmits the TTC information to each ROD through the VME P3 backplane. These data are then decoded and managed at ROD level by a TTCrx ASIC [18] and the TTC FPGA before they are transmitted to the PUs to synchronize the front-end and TTC data in the DSPs. On the other hand, the Busy signal is issued by the DSP when its input buffer is nearly full. Therefore there are 4 individual Busy signals in each ROD which are logically summed 'OR' in the VME FPGA. The result is transmitted to the TBM through the VME P3 backplane which collects the Busy status for all the RODs in the crate. Finally the TBM propagates the Busy status of the crate to the CTP via the ROD busy module and to the LTP to stop the generation of Level 1 Trigger acceptance signals, which would induce undesired dead-time in the detector.

- Control and monitoring flow: the VME slave core of the ROD is implemented in the VME FPGA. The VME bus is used to configure and read the status of the motherboard devices. Therefore, the control and monitoring of the board are performed through the VME bus.

The main components of the ROD module are presented in Figure 3.5 and their main functionalities are described in the following subsections.

3.5 DSP Code Structure

The kernel of the DSP code is programmed in standard C and is compiled and simulated using Code Composer Studio software provided by Texas Instruments [33]. Immediately after booting the DSP code the system is initialized and the EDMA ports, interrupt services and serial ports are disabled. Then, the DSP is configured. During this configuration the busy signal is set to avoid reception of data. Then, the input and output buffers are mapped with the default event size. Once the default configuration has finished, the DSP remains in an idle state until a command is received. At this point, new configuration commands can be asserted in order to reconfigure the DSP with proper values that will depend on the type of run (physics, Charge Injection, pedestals, laser, etc...). The Start command clears the busy signal, reinitializes the internal counters and allows reception of events.

The sequential execution of the processing starts upon the reception of a TTC event. The data from the detector is then synchronized with the TTC information, processed, packed and transmitted to the Output FIFO of the PU. The processing sequence is only paused to attend external interruptions or commands. These asynchronous events might produce discontinuities in the processing which were taken into account during the development of the kernel code.

The HPI provides access to read and write in the internal memory of the DSP. The writing through the HPI is used to change the configuration during data processing and to transmit the Optimal Filtering and calibration constants whereas the reading is used to extract information from the internal monitoring application.

Input and Output Buffers

The DSP code contains two input circular buffers (one per superdrawer) and one common output circular buffer. The maximum number of events inside the input buffers is calculated by the DSP at configuration time because it depends on the expected size of the events which

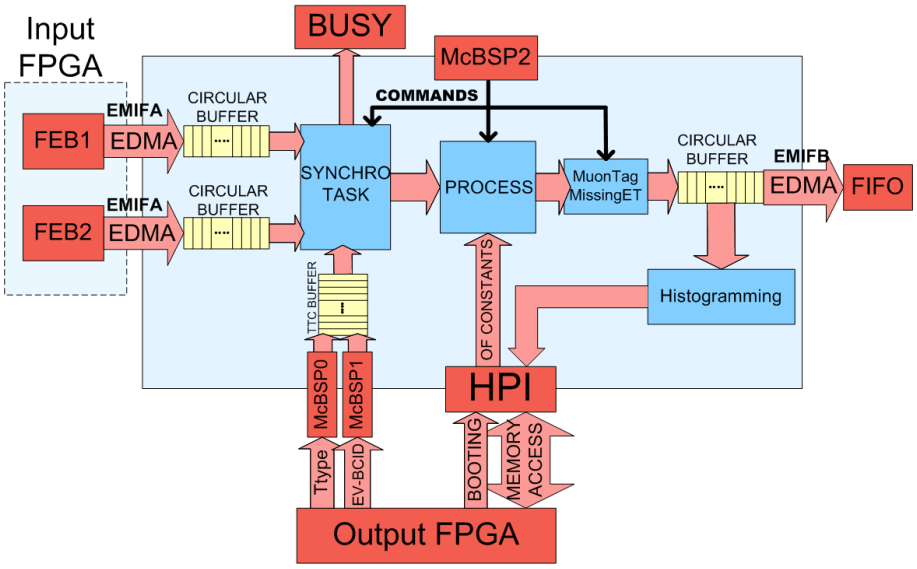


Figure 3.10: Block diagram of DSP code and data flow.

is different between physics and calibration runs. However, the minimum number of events is 16 which corresponds to the size of the output buffer of the superdrawer. In addition to the maximum number of events the DSP computes the addresses for the first and the last event in the buffer memory. Each time a new event is received, the DSP also computes the address where the event has to be stored by adding the event size to the previous event address. If the previous address plus the event size exceeds the buffer limit, the next event is stored in the first position implementing a circular buffer.

The data transfer between the Input FPGA and the DSP input buffers is performed through the EMIF A. This transfer is clocked at 100 MHz. There are two different interrupts from each Input FPGA in order to indicate which superdrawer is the data source. The DSP EDMA stores the received events in the input circular buffers. Once the event is reconstructed, it is copied to the DSP output circular buffer and transferred to the Output FIFO through the EMIF B that has a 16-bit bus width, and it is clocked at 100 MHz. The complete data flow and the different buffers are described in Figure 3.10.

In addition, there is a circular buffer to store the TTC information received through the

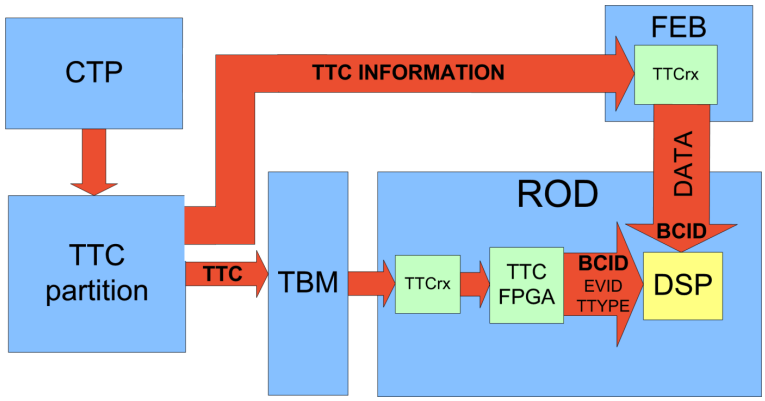


Figure 3.11: TTC distribution in TileCal readout chain.

McBSP0 and McBSP1 ports. The size of the buffer is fixed and can store up to 128 TTC event each of them composed by three 32-bit words containing the Event ID, BCID and TType.

Commands and Internal Registers

When the DSP is in the booted state, it is possible to send commands to it through the VME bus in order to change the configuration or the internal state. These commands are received through the McBSP2 port and the 8 LSBs correspond to the actual command field whereas the 24 MSB are the parameters for the command. The *configuration* command is used to set processing variables such as event size, processing function, operation mode, synchronization task as well as the activation of algorithms like the transversal energy sum of all the channels. The *prepare for run* command is used to set the variables of the run. This information, which is included in the header of each processed event by the DSP, and consists of the run number, data format version, source identifier, detector event type, sub-fragment header and the superdrawer identifier. As the space reserved in the command for arguments is limited, these values are read from a dedicated memory space which is accessed previously by the TDAQ software.

During data taking it is possible to readout some status registers in order to extract some information regarding the performance of the DSP. These registers are read through the HPI for convenience and the addresses of these registers are consecutive addresses starting from a

base address. The *probe* command updates the content of these registers.

Other commands are used to start and stop the data processing, to disable or enable the data inputs or to reset the counters and internal memory of the DSP.

Busy Handling and Synchronization Task

In order to avoid overwriting the input circular buffers, the DSP has to be able to stop the transmission of data from the front-end. Upon the reception of a data event, the DSP checks the number of events that are still to be processed. If this number if more than half of the buffer size (almost full flag), the DSP asserts the busy signal which is transmitted to the trigger to hold the generation of L1A signals. The number of non processed events in the input buffer is checked again with the transmission of each processed event to remove the busy status. The busy signal is implemented with hysteresis, therefore the busy is set and cleared at different values to avoid very short glitches in the busy signal. In particular, during operation each input buffer can store up to 24 events and the busy signal is asserted if there are more than 12 unprocessed events in the buffer and is removed with less than 12 events in the buffer.

The TTC events are received by the DSP much before the data arrives, as the same information is propagated from the local TTC system to the front-end and the back-end and it is assumed to be uncorrupted. Hence, the TTC events are always processed in the same order as received. For every TTC input event, the DSP checks the coincidence between the BCID in the TTC event and the BCID in the two input data events. If there is a coincidence, the events are processed. If there is no coincidence (BCID mismatch), the DSP tries to resynchronize both links searching in the events inside the input buffer. If no coincidence is found, the DSP tries to resynchronize at least one of the input data links with the TTC data. If one or both links stop sending data, the DSP will process the TTC event with null data events after a defined timeout. The time is given by a timer interrupt with programmable timer period. The complete state machine for the synchronization and busy process is detailed in Figure 3.12

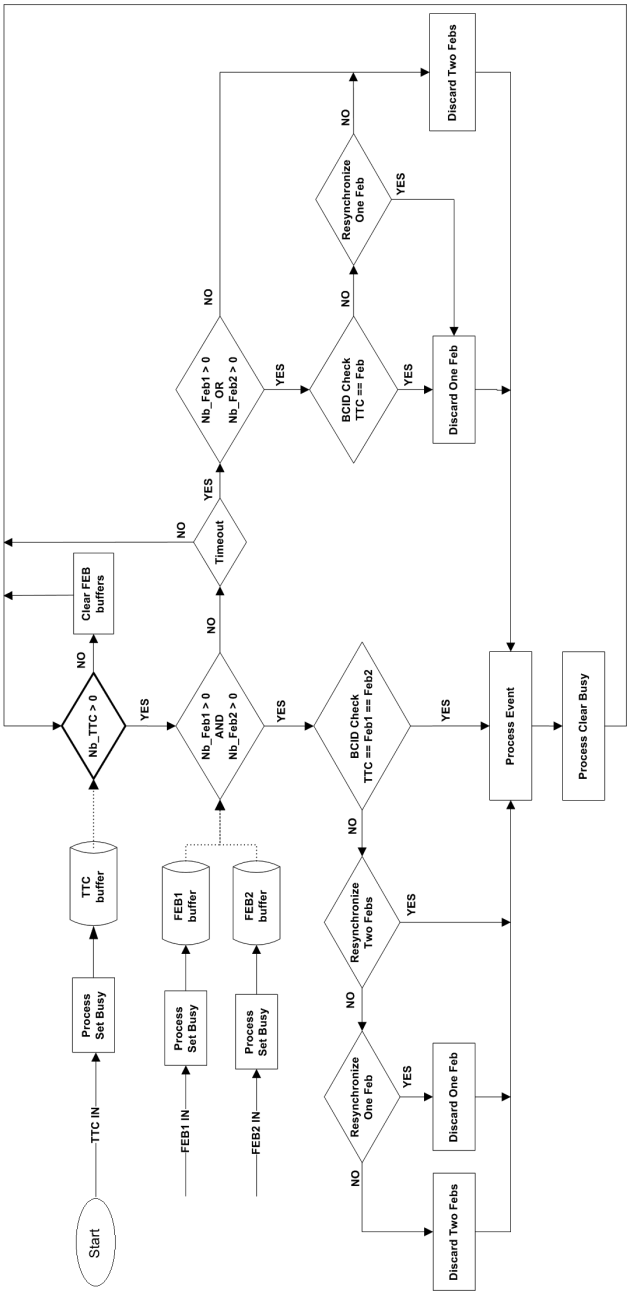


Figure 3.12: Synchronization and busy generation flow chart.

Processing Task

The processing task is the main functionality of the DSP code and it is executed always after the synchronization task. In absence of synchronization either due to BCID mismatch or timeout the processing task is executed with a null event. The first operation of the processing task is to complete the data quality fragment previously built in the Input FPGA. The processing task adds the BCID (not available in the InputFPGA) and the result of the synchronization task to flag BCID mismatches. Then, the processing task executes the signal reconstruction and other algorithms like the total transverse energy sum and the compression of digital samples, and finally the result is packed and transmitted to the DSP output buffer. During calibration runs the processing task is configured not to perform signal reconstruction algorithms and the front-end raw data is directly copied to the output buffer. Then, the raw data is reconstructed and analyzed offline. The physics mode includes different signal reconstruction algorithms configurable through the TDAQ software at the beginning of the each run. It includes different flavors of the Optimal Filtering method as well as the total transverse energy sum algorithm. A complete description of these algorithms can be found in Chapter 6.

Raw data compression Apart from the signal reconstruction performed in the DSP, a portion of the digital samples are transmitted for offline analysis. In particular, the DSP selects and compresses the samples for channels with an amplitude above a configurable threshold in the so-called Fragment 1. In this case the amplitude is considered as the difference between the maximum and minimum samples of the pulse. There are two versions of Fragment 1 implemented in the DSP.

- Version A. If version A is used the samples of selected channels above the threshold are copied in the output fragment without any data compression. This version was used in the early data taking period were the Level 1 Trigger rates were considerable lower than the nominal values.

- Version B. On the other hand, the version B applies a data compression to reduce the fragment size. In this case, samples of the HG channels with an amplitude between

the configurable threshold and the hard-coded threshold of 15 ADC-counts (4 bits) are packed in the so-called Format 1. The Quality Factor of the reconstruction for these type of signals is always zero. Therefore, in order to save processing time the Quality Factor is not computed for channels selected as Format 1 (low amplitude signals). LG signals and signals above the 15 ADC-counts threshold are packed in the so-called Format 2. Finally, the samples of channels below the configurable threshold are not transmitted in order to reduce the size of the output fragment.

The size of the Fragment 1 depends on the occupancy of the detector depositions. The configurable threshold is used to prevent a saturation in the output of the ROD. During the last data-taking period of 2012 (before the Long Shutdown 1 of the LHC) the version B of Fragment 1 was used with a threshold of 5 ADC-counts (approximately 60 MeV) for a peak luminosity of 7.3×10^{33} cm^{-2}s^{-1} and $< \mu >$=30 and no bandwidth saturation was observed.

3.6 DSP Monitoring and Histogramming

The DSP monitoring samples every event passing the Level 1 Trigger selection (Figure 3.13). It aims to spot problems related to the data acquisition process. The information is stored in the DSP internal memory and can be retrieved by a collector application at any time.

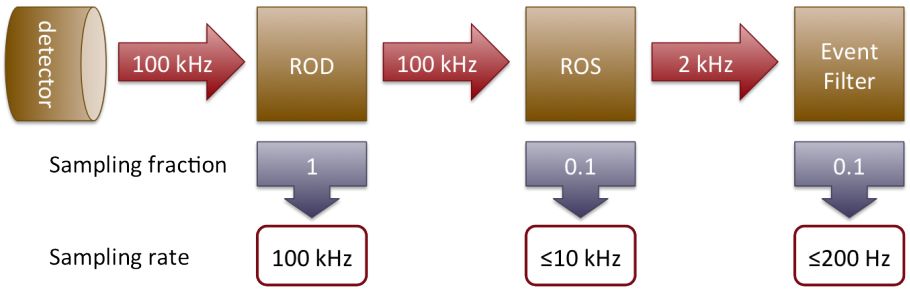


Figure 3.13: Summary of the different Tile Calorimeter monitoring levels and rates.

There are two types of results, the data-flow counters (RITMO) [34] and the histograms. The data-flow counters that are available provide statistics about processed events. The histograms are produced inside the DSP and contain information about Optimal Filtering quality factor and first sample distributions for all events.

RITMO

The ROD Information for Tile Monitoring (RITMO), is a selection of counters from the DSPs that are published on every probe.

There are three types of counters (*in, out, discarded*) that count the number of events per superdrawer, thus 6 counters per DSP. The number of events *in* is incremented each time a data fragment from the superdrawer arrives to the DSP input memory buffer. The number of events *out* counts the number of events that have been processed from the input buffer and sent to the output buffer. The number of events *discarded* counts the number of unsuccessful trials to process data from the input buffer, either because the data was missing or due to a mismatch between the BCID of the data and the TTC event. The number of TTC events *in* counts the number of TTC events received by the DSP through the backplane of the ROD crate from the TTC system. These events drive the processing of the DSP, no event is processed if no TTC events are received. The number of *total* events out is the number of processed events given by the number of TTC events received. The number of TTC events *out* counts the number of TTC events that have been sent to the output. The busy of the DSP is monitored by the instantaneous *busy status* register, which can only take values 0 or 1, and the *number of busy* counts, which counts the number of times the DSP has been busy. The conditions that could cause the DSP to assert the busy are related to burst of data in the input buffer or output data bandwidth limitations. There are also three counters that are overwritten with every TTC event, the aim of which is to sample the last TTC event processed in the case the trigger is paused. These counters are the Extended L1ID, BCID and Trigger type.

3.7 Trigger and Data Acquisition Software

The ATLAS TDAQ software sub-system encompasses all the software to do with configuring, controlling and monitoring the data acquisition system, but excludes anything to do with the management, processing or transportation of physics data [35]. It is essentially the glue that holds the various sub-systems together. It does not contain any elements that are detector specific as it is meant to be used by all possible configurations of the DAQ and detector ins-

trumentation. It coexists and cooperates with the other sub-systems; in particular, interfaces that are required to the data flow, triggers, processor farm, event builder, detector readout crate controllers and DCS. The various components of the Online Software include:

- Inter Process Communication (IPC): The core communication service. It relies on the underlaying TCP/IP message passing.
- Information Services (IS): The service that allows sharing of any kind of user defined information between applications.
- Configuration databases: The implementation of the database system used to describe the configuration. It includes the description of all the hardware modules.
- Process Manager: The service responsible for the execution of applications on the nodes.
- Access Manager: The service responsible for the access to resources and nodes for a given user.
- Resource Manager: The bookkeeping of allocated resources.
- Run control: The core Finite State Machine (FSM) service responsible for the structure of controlled applications.
- Monitoring: The online monitoring infrastructure.
- Integrated Graphical User Interface (IGUI): The user interface with the run control.
- Expert System: The service that supervises the recovery actions.

3.7.1 ROD Crate DAQ

The ROD Crate DAQ (RCD) framework is an extension of the ROS software that offers the possibility to integrate the handmade hardware into the general TDAQ software [11]. The RCD is based on a multi-threaded readout application core which loads specific plug-ins at runtime to adapt its behavior to the detector specific needs. There are different plug-ins for different purposes. A schematic overview of the readout application is shown in Figure 3.14.

The RCD application is commanded by the run control service. When a run control transition command is sent by the user through the IGUI, a request for a state transition is sent to the RCD. The RCD reacts to the request and operates the controlled hardware according to the detector specific code. The relationships between the TDAQ run control

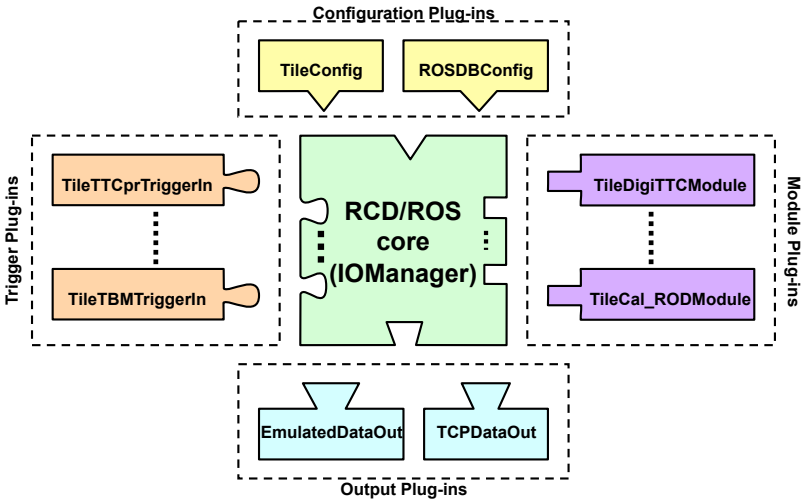


Figure 3.14: ROD Crate DAQ schematic diagram.

FSM transition commands and RCD methods have been detailed before in [35].

The RCD has four different plug-in types:

- The Configuration plug-in loads the information from the configuration database and passes it to the RCD core application.
- The Trigger plug-in handles the trigger requests for data fragments. It runs as a separate thread that queues requests on a request list. A number of request handlers from the RCD dequeues them by executing the data requests that push the output information to the output plug-in.
- The Module plug-in is a controlled resource which receives transition commands from the RCD. In ATLAS operation mode it does not send data out to the output plug-in.
- The Output plug-in manages the output of the data which can be sent elsewhere through ethernet or written down to a file.

In particular, a module plug-in describes a hardware or software component that is controlled by the RCD. It can extend any of the methods of the ReadoutModule class which represent a run control transition command.

TileCal specific module plug-ins for their use in the RCD framework are :

- TileDigiTTCModule. This module is a double purpose module. It controls the front-end electronics through 3-in-1 TTC commands and the back-end electronics needed for TTC.
- TileVMEReadoutModule. It is a general purpose module. In calibration runs it is configured to readout data from the shared memory region of the crate controller and provide data fragments to the output plug-in.
- TileOfcShameModule. It is a plug-in module that does not control any hardware module. It is used as the conditions data cache for the whole crate. In order to avoid eight connections to the conditions database, one per ROD module, this module is used to gather conditions settings and store it in the shared memory ("shame") region of the crate controller.
- TileShaftModule. It is a plug-in to control the Shaft board, which is a specific VME board to share the calibration requests in empty bunches. This module controls the different calibration trigger requests foreseen for TileCal. Each calibration trigger request can be individually enabled and delayed with respect to the signal that marks the turn of the LHC orbit clock.
- TileLaserModule. This plug-in module controls the Laser ROD. This is a ROD which does not readout data from the front-end but from the back-end, it is used in physics runs to insert calibration information into the data-flow. Its readout link is connected to the LBC partition readout system. It receives TTC information from the shafts and the calibration information of the current event is readout from IS.
- TileCal_RODModule. It controls the ROD motherboard. This module is described in next Section.

3.7.2 The ROD Module

The TileCal_RODModule configures the ROD motherboard and the PUs according to the run type [34]. For physics runs, ROD is configured as a data-flow element between the front-end

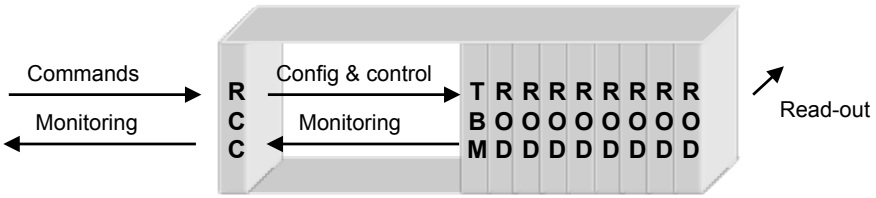


Figure 3.15: Schematic view of the ROD crate operation in S-link readout mode. Commands are issued by the user to the ROD Crate Controller (RCC), and monitoring quantities are returned from it. The configuration and control of the boards is done through the VME bus. Monitoring quantities are readout from the boards. Data is trasmitted to the ReadOut System through S-Link.

electronics and the readout system. Figure 3.15 shows the schematic overview of the operation of the ROD crate. It takes data in from the front-end links and sends data out through the ROLs. Monitoring quantities computed by the boards are accessed from the crate controller and made available to the TDAQ monitoring service.

Eight instances of the TileCal_RODModule are needed by the RCD application in the ROD crate. Each one controls a different motherboard independently. Masking of individual input links to the motherboard may be done through the TileCal specific IGUI panel in the TDAQ software. The online processing algorithm may be configured through the Globals sub-panel of the Tile IGUI panel, shown in Figure 3.16. The conditions settings are retrieved accordingly at configuration time, in the *prepare-for-run* transition, from the shared memory region of the crate controller.

3.8 ROD Data Format

The format of the data fragment provided by the RODs is specific to each sub-detector as detailed in the ATLAS raw event format [36]. It has a modular structure with four types of elements: Header, data elements, status words and trailer. From the readout point of view, a ROD fragment is the single packet of data transferred per event through each ROL from the ROD to the ROS. In the TileCal case, each ROD fragment contains the data from four adjacent superdrawers. In the following chapters we will first describe the TileCal ROD input data format which is completely defined by each sub-detector. Then, we will describe the

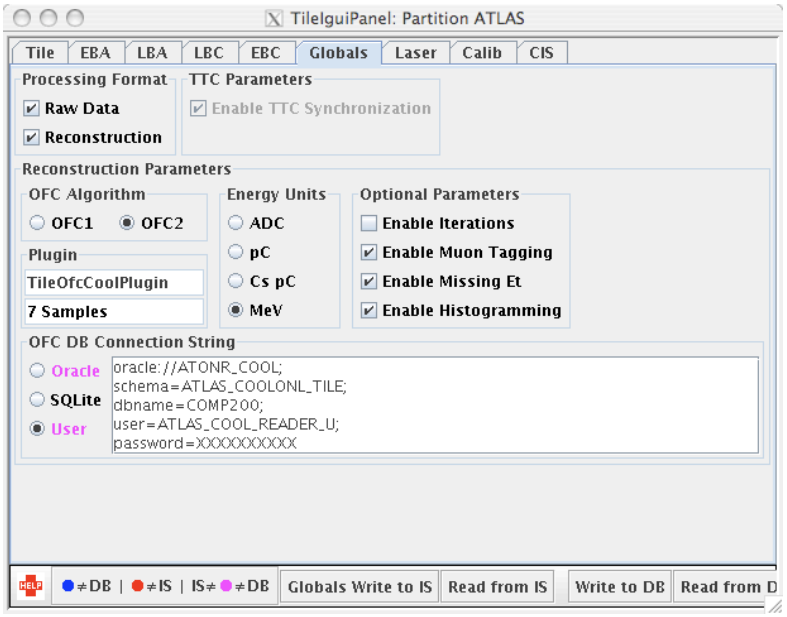


Figure 3.16: Globals sub-panel in the Tile IGUI panel.

different output sub-fragments and the combination of the sub-fragments according to the various operation modes.

3.8.1 ROD Input Data Format

The front-end electronics can be configured to transmit up to 16 samples and 2 gains per channel for each selected event. However, currently it is always configured to transmit 7 samples for each channel. The auto-gain mode is used for physics runs (the low gain is transmitted if at least one ADC sample is overflow in the high gain range) whereas for calibration runs both high and low gains are transmitted (bi-gain mode). The ratio between high and low gain is 64. Although the number of instrumented channels in Extended and Long barrels is different, the same data format is used to have a well balanced system. Table 3.2 shows the general Interface Board output data format which corresponds to the ROD input format. Every data event is composed by a *Header* used to signal the start of the event, 16 DMU data blocks including the data from 3 PMTs each, a *DMU chip mask* word, a *Global CRC* word and a *Trailer* to indicate the end of the event.

Header - Start of Event
DMU1 Data Block
DMU2 Data Block
.....
DMU16 Data Block
DMU Chip Mask Word
Global CRC
Trailer - End of Event

Table 3.2: TileCal raw data event format. DMU data blocks depend on the operation mode (Tables 3.3 and 3.4).

Header and Trailer The Header word transmitted by the Interface Board is 0x51115110 and the Trailer word is 0xFFFFFFF0. However, the version of the G-link receiver chip used in the ROD only allows 14-bit control words. The Header word is replaced at the very input of the ROD by 0x11111110 and the Trailer by 0x3FFF3FF0.

The DMU data block The 16 DMU data blocks within the event corresponds to the 16 DMU chips mounted in the superdrawer, each containing the data for three channels. In the Extended Barrel only 12 DMUs are present but the other four DMU data blocks are filled with zeros by the Interface Board. So that, the data format does not vary due to non instrumented channels. The DMU data block is different for physics runs, where only one gain is transmitted (Table 3.3) and for calibration runs, where the two gains are sent (Table 3.4).

The first word in the DMU data block is the DMU Header which contains information about the configuration used, digital checks and the gain selected for each channel (Table 3.5). Following, the data words contain the samples for three channels plus a parity check. There are as many data words as samples in each DMU block. The PMT number packed

32-bit words				
DMU Header				
0	P	Sample1 Ch3	Sample1 Ch2	Sample1 Ch1
0	P	Sample2 Ch3	Sample2 Ch2	Sample2 Ch1
0	P	Sample3 Ch3	Sample3 Ch2	Sample3 Ch1
0	P	Sample4 Ch3	Sample4 Ch2	Sample4 Ch1
0	P	Sample5 Ch3	Sample5 Ch2	Sample5 Ch1
0	P	Sample6 Ch3	Sample6 Ch2	Sample6 Ch1
0	P	Sample7 Ch3	Sample7 Ch2	Sample7 Ch1
DMU CRC word				

Table 3.3: DMU data block in auto-gain (physics) mode.

32-bit words				
DMU Header (LG)				
0	P	LG Sample1 Ch3	LG Sample1 Ch2	LG Sample1 Ch1
0	P	LG Sample2 Ch3	LG Sample2 Ch2	LG Sample2 Ch1
0	P	LG Sample3 Ch3	LG Sample3 Ch2	LG Sample3 Ch1
0	P	LG Sample4 Ch3	LG Sample4 Ch2	LG Sample4 Ch1
0	P	LG Sample5 Ch3	LG Sample5 Ch2	LG Sample5 Ch1
0	P	LG Sample6 Ch3	LG Sample6 Ch2	LG Sample6 Ch1
0	P	LG Sample7 Ch3	LG Sample7 Ch2	LG Sample7 Ch1
DMU Header (HG)				
0	P	HG Sample1 Ch3	HG Sample1 Ch2	HG Sample1 Ch1
0	P	HG Sample2 Ch3	HG Sample2 Ch2	HG Sample2 Ch1
0	P	HG Sample3 Ch3	HG Sample3 Ch2	HG Sample3 Ch1
0	P	HG Sample4 Ch3	HG Sample4 Ch2	HG Sample4 Ch1
0	P	HG Sample5 Ch3	HG Sample5 Ch2	HG Sample5 Ch1
0	P	HG Sample6 Ch3	HG Sample6 Ch2	HG Sample6 Ch1
0	P	HG Sample7 Ch3	HG Sample7 Ch2	HG Sample7 Ch1
Not used				

Table 3.4: DMU data block in bi-gain (calibration) mode. In this case two data blocks containing the low and high gain of samples are always transmitted.

in each DMU block is different for the Long and Extended barrels. Tables 3.6 and 3.7 show the mapping of the channel to the cell PMT including the non instrumented channels. The transmission of the DMU data block from the DMU chips to the Interface Board is performed through two serial lines used to separately transmit even and odd bits. At the end of the transmission of each DMU data block the DMU chip appends the result of a CRC computation

31	**30**	**29 .. 26**	**25**	**24**	**23**	**22**	**21 .. 18**	**17**	**16,15**	**14 .. 12**	**11 .. 0**
H	P	L	E	S	D	R	V	'0'	M	Gain	BCID

Note:
BCID. Bunch Crossing Identification for the present event.
Gain. One bit to indicate the gain for each 3-in-1 card (0=low , 1=high).
M . Mode (00=normal ; 01=calibration ; 10 = Test ; 11=not used).
V . Variable parity: parity from the variables in the chip.
R . Register parity: parity from the registers in the chip.
D . Double Strobe Error received from the TTC.
S. Single Strobe Error received from the TTC.
E. Parity error from internal memory during last readout.
L. Derandomizer length (number of samples).
P. Parity (odd).
H. DMU header identifier (0x1).

Table 3.5: Data format of the DMU Header word.

for even (16 MSB) and odd (16 LSB) bits. In the bi-gain mode the CRC is not computed and an empty word is included at the end of the data block.

***DMU Chip Mask* word** The CRC for even and odd bits of each DMU data block is again computed at the Interface Board. The result is compared with the CRC appended by each DMU chip at the end of the DMU data block. The result of the comparison is packed in the *DMU Chip Mask* word. The number of bits with logical value '1' in the 16 LSB indicates the number of DMU data block with correct CRC for odd bits whereas in the 16 MSB corresponds to even bits. In the case of non instrumented digitizers the CRC check is flagged as error.

***Global CRC* word** The Interface Board also computes a CRC for the complete event and the result is appended in the 16 LSB of the *Global CRC* word. This CRC can be used in the back-end electronics to detect errors in the data transmission. This information can be used by the OMB to select which of the redundant data packets is free of errors and therefore should be transmitted to the ROD.

3.8.2 The DSP Input Data Format

The data transmitted from the Interface Board to the ROD through the optical link are received in the ORx and then deserialized to 16-bits words in the G-Link chips at 40 MHz.

DMU	Ch	PMT	DMU	Ch	PMT	DMU	Ch	PMT	DMU	Ch	PMT
	0	1		3	4		6	7		9	10
1	1	2	**2**	4	5	**3**	7	8	**4**	10	11
	2	3		5	6		8	9		11	12
	12	13		15	16		18	19		21	22
5	13	14	**6**	16	17	**7**	19	20	**8**	22	23
	14	15		17	18		20	21		23	24
	24	27		27	30		30	33		33	36
9	25	26	**10**	28	29	**11**	31	32	**12**	34	35
	26	25		29	28		32	31		35	34
	36	39		39	42		42	45		45	48
13	37	38	**14**	40	41	**15**	43	44	**16**	46	47
	38	37		41	40		44	43		47	46

Table 3.6: Mapping of channel to PMT for DMU index in a Long Barrel module. Note that underlined PMT number corresponds to non instrumented channels.

DMU	Ch	PMT	DMU	Ch	PMT	DMU	Ch	PMT	DMU	Ch	PMT
	0	1		3	4		6	7		9	10
1	1	2	**2**	4	5	**3**	7	8	**4**	10	11
	2	3		5	6		8	9		11	12
	12	13		15	16		18	19		21	22
5	13	14	**6**	16	17	**7**	19	20	**8**	22	23
	14	15		17	18		20	21		23	24
	24	27		27	31		30	33		33	36
9	25	26	**10**	28	32	**11**	31	29	**12**	34	35
	26	25		29	28		32	30		35	34
	36	44		39	43		42	45		45	48
13	37	38	**14**	40	42	**15**	43	39	**16**	46	47
	38	37		41	41		44	40		47	46

Table 3.7: Mapping of channel to PMT for DMU index in an Extended Barrel module. Note that underlined PMT number corresponds to non instrumented channels.

The deserialized 16-bits words, including control words for G-Link protocol synchronization, are transferred to the Staging FPGA. The Staging FPGA discards the control words of the protocol and transmits only the data event to the Input FPGA. In principle, the Staging FPGA transmits the data words included between the reception of a Header and a Trailer word. The Input FPGA includes two operation modes in terms of data flow. In the calibration mode, the received events are directly transferred to the DSP and therefore the DSP input

31	**....**	**16**	**15**		**....**	**0**
0	Sample1 **CH1** [25 .. 16]		G [15]	0	BCID[11 .. 0]	
0	Sample3 **CH1** [25 .. 16]		0		Sample2 **CH1** [9 .. 0]	
0	Sample5 **CH1** [25 .. 16]		0		Sample4 **CH1** [9 .. 0]	
0	Sample7 **CH1** [25 .. 16]		0		Sample6 **CH1** [9 .. 0]	
0	Sample1 **CH2** [25 .. 16]		G[15]	0	BCID[11 .. 0]	
0	Sample3 **CH2** [25 .. 16]		0		Sample2 **CH2** [9 .. 0]	
0	Sample5 **CH2** [25 .. 16]		0		Sample4 **CH2** [9 .. 0]	
0	Sample7 **CH2** [25 .. 16]		0		Sample6 **CH2** [9 .. 0]	
.....						
0	Sample1 **CH48** [25 .. 16]		G[15]	0	BCID[11 .. 0]	
0	Sample3 **CH48** [25 .. 16]		0		Sample2 **CH48** [9 .. 0]	
0	Sample5 **CH48** [25 .. 16]		0		Sample4 **CH48** [9 .. 0]	
0	Sample7 **CH48** [25 .. 16]		0		Sample6**CH48** [9 .. 0]	

Table 3.8: DSP input channel format in Physics mode.

data format corresponds to the ROD input data format detailed in the previous Section. On the other hand, in the physics operation mode the Input FPGA performs a re-formatting of the input data to optimize the data processing in the DSP. Essentially, the information included in the DMU data block (Table 3.3) is extracted and packed into two different new fragments.

Channel Format The format based on DMU data blocks is transformed in a channel based format. In the new format the samples of the 48 channels are extracted and packed separately for each channel. The 10-bit samples are packed in the 16 LSB and 16 MSB of 32-bit words (Table 3.8). A total of 4 32-bit words per channel are used to pack the 7 samples, the gain and the BCID of the DMU block.

Data Quality Fragment The Data Quality (DQ) Fragment is build in the Input FPGA and transferred to the DSP just after the Channel samples. It essentially contains the information of the 16 DMU data blocks Headers, therefore the information is packed in 16-bit words. Moreover, the Input FPGA computes the Global CRC, the DMU CRC (even and odd bits separately) and the parity and the results are compared with the computation performed in the front-end. Any mismatch is flagged in the Data Quality Fragment to mask corrupted

data. Table 3.9 shows a complete description of the Data Quality Fragment data format. The DQ fragment is slightly updated in the DSP and then it is included in the ROD output data fragment. On the one hand, the 16 MSB of the first word (not used in the DQ Fragment built in the Input FPGA) are used to indicate a discarded event (bit(31) = '0') and to transmit the TTC BCID (bits [27..16]). On the other hand, the second bit of the BCID word (corresponding to the second DMU block) is set to '0' if the data event is correctly synchronized with the TTC event (BCID coincidence between second DMU and TTC). Table 3.10 shows the expected DSP output DQ Fragment when there are no digital errors in the data. The expected fragment is different for each partition and for special modules in Extended Barrel because non-instrumented DMU chips are signaled as errors in the DQ fragment.

3.8.3 ROD Output Data Format

The ATLAS ROD output fragment is specific to each sub-detector according to the raw event format [36]. It is a modular structure with four types of elements: Header, data elements, status words and trailer (Table 3.11). From the readout point of view, a ROD fragment is the single packet of data transferred for each event from the ROD to the ROS via a ROL. Therefore, it includes the data processed in one ROD PU (2 DSPs or 4 superdrawers). The data received in the DSP from the front-end electronics is processed, formatted and transmitted to the Output Controller of the ROD where the ROD fragment is built according to the specific TileCal data format. The upper DSP (Header) in the PU is responsible for building the ROD fragment Header which is used to identify the fragment in the subsequent levels of trigger. The data elements correspond to the actual information from the detector which in the case of TileCal includes information from 4 superdrawers in each ROD fragment. The status words are not used in TileCal and finally the Trailer word is used to detect the end of fragment at ROS level.

Header The ROD header is partially built by the OC and the DSP Header. The OC adds a 32 bit control word before the header block to indicate to the HOLA card the Beginning of Fragment according to the S-Link protocol [23]. This word is not transmitted to the ROS and it is only used for internal transmission between the ROD and the HOLA card in the TM.

31 16	**15 .... 0**
0	Global CRC
Memory Parity	BCID flag
Double Strobe	Single Strobe
Header Parity	Header bit 31
Samples Parity	Samples bit 31
ROD *DMU Chip Mask*	FE *DMU Chip Mask*

Note:
Global CRC. The LSB of the word is set to '1' if there is a mismatch between the Global CRC computed in the Input FPGA and in front-end.
Memory Parity. The Memory Parity Error bit of each DMU data block Header is copied in the corresponding position in the 16-bit word ('1' = error).
BCID flag. The BCID of the second DMU data block is compared with all the rest. A mismatch is flagged in the corresponding bit ('1' = error). The second bit, corresponding to the second DMU block used as reference, is updated in the DSP with the result of the comparison between its BCID and the TTC BCID.
Single and Double Strobe. Both Single and Double Strobe error bits are copied from the DMU Header block. They correspond to Strobe errors in the TTCrx chip of the digitizer.('1' = error).
Parity. The Input FPGA computes the even parity for every 32-bit word received and the result is compared with the value included in the data in the front-end. The result is flagged in the Header and Samples parity words. ('1' = error).
Bit 31. The MSB (bit '31') is used to indicate a Header ('1') or a samples ('0') word. The Input FPGA checks the value of this bit and if any error if found in a DMU block it is flagged in the corresponding bit. ('1' = error).
DMU Chip Mask. The CRC for even and odd bits of each DMU block is computed and compared with the result included in the data. The result for each DMU is stored in the corresponding bit of the ROD *DMU Chip Mask* word. The FE *DMU Chip Mask* 16-bit word is built as a logical OR of the two bits (front-end CRC check for even and odd bits for each DMU) of the *DMU Chip Mask* 32-bit word received from front-end. Note that the logic is different in these two words. ('0' = error).

Table 3.9: Data format of the Data Quality fragment computed in the Input FPGA.

The actual Header (excluding the Beginning of Fragment word) is derived from the ATLAS event format header (Table 3.12) and includes:

- Start of header marker: Each fragment starts with a header marker, in the ROD case it is 0xEE1234EE. The asymmetry of the word allows the identification of the byte ordering in the ROD data.

Long Barrel	Extended Barrel	Special Cases
32-bit words	**32-bit words**	**32-bit words**
0x8*xxx*0000	0x8*xxx*0000	0x8*xxx*0000
0x00000000	0x0000C300	0x00000C31
0x00000000	0x00000000	0x00000000
0x00000000	0xC300C300	0x0C310C31
0x00000000	0xC3000000	0x00000C31
0xFFFFFFFF	0x3CFF0FFF	0x3CFE07FF

Table 3.10: Expected ROD output DQ Fragment. Due to non-instrumented DMU chips the expected fragment is different for modules in Long Barrel (left), in Extended Barrel (middle) and the two special cases EBA15 and EBC18 (right). Note that *xxx* corresponds to the TTC BCID included for each event by the DSP.

ROD Fragment Header
Data Elements
Status words
ROD Fragment Trailer

Table 3.11: ATLAS ROD fragment general structure.

- Header size: Number of 32-bit words in the header including itself. For ATLAS operation it is 9.
- Format version number: It is composed by two 16-bit fields corresponding to data format and DSP code versions.
- Source identifier: uniquely indicates the origin of the fragment across all detectors. It is created by the DSP and it is composed by sub-detector (Table 3.13) and ROD module identifier.
- Run number. This is a 31-bit number incremented at the beginning of each run in ATLAS. It is retrieved from a common database and copied to the DSP during configuration state.
- Extended Level 1 Identifier: The 8 most significant bits compose the Event Counter

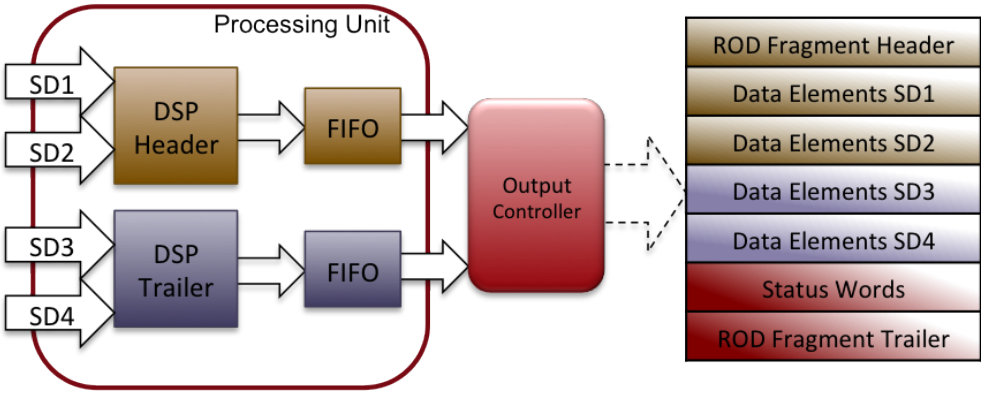


Figure 3.17: Structure of the PU output data format. The status elements and trailer of the fragment are added in the OC before the transmission to the ROS.

Byte	**3**	**2**	**1**	**0**
Start of header marker	0xEE1234EE			
Header size	0x9			
Format version number	Major Version		DSP Version	
Source ID	reserved	Sub-det	Module ID	
Run number	0	run number 31 bits		
Extended L1ID	ECRID	L1ID		
BCID	0		0	BCID 12-bit
Trigger Type	0			TType
Detector Event Type	Fragment type mask			Event type

Table 3.12: TileCal ROD fragment header.

Reset Identifier (ECRID) and the remaining 24 bits encode the L1ID.

- Bunch crossing: The bunch crossing identifier is coded as a 12-bit word in the LSB of the 32-bit word. The 20 upper bits are not used and set to zero.
- Level 1 Trigger Type: 8-bit word sent by the TTC system and mounted in the header. The 24 upper bits are not used and set to zero.
- Detector event type: Each sub-detector can define a detector event type. For TileCal it is divided in two parts: the lower byte is the Tile event type (Table 3.14) and the upper 24 bits is the fragment type mask.

ID	Source
0x51	Long barrel A side
0x52	Long barrel C side
0x53	Extended barrel A side
0x54	Extended barrel C side
0x50	Back-end

Table 3.13: TileCal sub-detector ID values.

Value	Description
0	Unknown
1	Physics
2	LED/Laser
4	Pedestals
8	CIS mono
16	CIS scan

Table 3.14: TileCal run type values.

Data elements The TileCal ROD data elements, also called ROD sub-fragments, have a modular structure. Each data element consists of a header followed by a body. The sub-fragment header contains a start of header marker, the fragment size (number of 32-bit words including the 3 words of the header) and a source identifier (Table 3.15). The Source Identifier is divided into module identifier, which is used to recognize the front-end module from which the data was received (Table 3.16); the sub-fragment identifier, which specifies the type of data contained in the block (Table 3.17); and extra information to specify some parameters of the corresponding algorithm used to build the data element.

A complete description of the different data elements and the possible combinations according to the operation mode will be described in Section 3.8.4.

Byte	3	2	1	0
Start of header marker	0xFF1234FF			
Sub-Fragment Size	Number of 32-bit words			
Source ID	Extra Info	Sub-Frag ID	Module ID	

Table 3.15: ROD fragment data element header.

Partition	Module ID
LBA	0x100 - 0x13F
LBC	0x200 - 0x23F
EBA	0x300 - 0x33F
EBC	0x400 - 0x43F

Table 3.16: Module identifier.

Data element	Sub-fragment ID
Raw data	0x00
Compressed raw data	0x01
Reconstruction	0x04
Data Quality	0x0A

Table 3.17: Sub-fragment identifier.

Status Elements The status elements are used to flag the status of the fragment. This element is included by the OC of the ROD for each fragment. TileCal does not use this element and it is always transmitted empty. However, in TileCal the Data Quality Fragment is used to flag digital errors in the data. This is included as an extra data element for each module.

Trailer The Trailer contains the number of data elements, the number of status elements and the status block position. The current ROD data format does not implement status elements. Therefore, the number of status elements is set to zero. The value of the status block position defines where the status elements are located; value zero indicates the status block precedes the data elements and value one indicates the status block follows the data block. The value of the status block position is arbitrarily set to one.

3.8.4 Data Elements Description

The TileCal data elements, also called sub-fragment types, include the result of the data processing inside the ROD. There are four different fragment types corresponding to different online algorithms implemented in the ROD.

Raw Data Sub-Fragment

The raw data fragment type contains non processed data as received from front-end (Table 3.2) but without the Header and and Trailer words. This fragment can be used offline to reconstruct the samples at any time. However, this operation mode can not be used at the nominal Level1 trigger rate because it requires a high output data bandwidth. The size of the fragment depends on the number of samples (currently 7 samples are always used) and the gain mode used in the front-end (Equation (3.1)). The usual size for auto-gain mode is 146 32-bit words whereas it is 274 32-bit words for bi-gain mode used in some calibration runs. The fragment is completed with the three 32-bit words in the sub-fragment header. The sub-fragment ID included in the sub-fragment header is 0x00 (Table 3.15) and the Extra Info field is not used.

The raw data sub-fragment is used for calibration runs where the data is analyzed offline and the Level 1 Trigger rate is considerably lower than in physics mode. Moreover, the raw data sub-fragment was used during commissioning period and early data taking when the LHC peak luminosity and therefore the Level 1 Trigger rate were lower than the nominal values. In that period the raw data was transmitted together with the DSP reconstruction fragment (Section 3.8.4) allowing the validation of the online reconstruction by using the offline reconstruction of the raw data as reference.

$$Size = ((((Samples + 1) \times Gains) + 1) \times 16) + 2 \tag{3.1}$$

Compressed Raw Data Sub-Fragment

Due to limited ROD output data bandwidth when running at high Level 1 rate conditions it was implemented a compressed version of the raw data fragment to allow the offline reconstruction of channels with signal above a programable threshold. Hence, the size of the fragment varies event by event depending on the number of channels that accomplish the conditions. The digits of channels in HG with a difference between maximum and minimum sample above a programmable threshold or in LG are included in this fragment. There are two versions of the fragment where only the packing format of the digits changes. In the first version four 32-bit words are used to pack the samples of each selected channel (Table 3.18). It

31	**....**	**16**	**15**		**....**	**0**
0	Sample1 **CH1** [25 .. 16]		G [15]	0	**Module ID**[11 .. 0]	
0	Sample3 **CH1** [25 .. 16]		0		Sample2 **CH1** [9 .. 0]	
0	Sample5 **CH1** [25 .. 16]		0		Sample4 **CH1** [9 .. 0]	
0	Sample7 **CH1** [25 .. 16]		0		Sample6 **CH1** [9 .. 0]	

Table 3.18: Data format for compressed raw data format for one channel.

31 ... 26	**25 ... 16**	**15 ... 12**	**11 ... 8**	**7 ... 4**	**3 ... 0**
ChID	S_{min}	0	$S_7 - S_{min}$	$S_6 - S_{min}$	$S_5 - S_{min}$
		$S_3 - S_{min}$	$S_4 - S_{min}$	$S_1 - S_{min}$	$S_2 - S_{min}$

Table 3.19: Compressed raw data format for low amplitude signals (6 bytes format).

essentially does not apply any compression and it just selects the channels above the threshold and the samples are copied in the output fragment replacing the BCID in the first word by the channel ID. This version was used during early operation of ATLAS at low instantaneous luminosity and Level1 trigger rate with respect to nominal values. In order to avoid ROD bandwidth saturation with the expected increase in the instantaneous luminosity of the LHC during 2012, it was implemented a more optimized version of this fragment. In this case, depending on the amplitude of the signal a different packing format is used. First, the digits of channels in HG and with a difference between minimum and maximum samples above the programmable threshold but below 15 ADC-counts are packed using 6 bytes (Table 3.19) whereas the channels above 15 ADC-counts or in LG are packed using 10 bytes (Table 3.20).

The channels packed in the 6 bytes format are located in the first part of the sub-fragment and the number of channels packed with this format is included in the extra info field in the header. The fragment is unpacked using the total size of the sub-fragment included in the header and the number of channels in the 6 bytes format. Moreover, the MSB in the extra info field is used to indicate the compressed raw data format used; if no compression is applied (former packing format) the MSB is '0' whereas it is '1' for the optimized format. Table 3.21 shows an example of the optimized compressed raw data sub-fragment with 3 channels in the 6 bytes format and 2 channels in the 10 bytes format.

31			...		**0**
S_3^{bottom}[31..27]	S_2[26..17]	S_1[16..7]	G[6]	ChID[5..0]	
S_6^{bottom}[31..25]	S_5[24..15]	S_4[14..5]		S_3^{top}[4..0]	
		0	S_7[12..3]	S_6^{top}[2..0]	

Table 3.20: Compressed raw data format for high amplitude signals (10 bytes format).

Byte	**3**	**2**	**1**	**0**
Start of header marker	0xFF1234FF			
Sub-Fragment Size	13			
Source ID	0xB	0x01	Module ID	
DATA	Channel 01		Channel 01	
	Channel 15		Channel 01	
	Channel 15		Channel 15	
	Channel 33		Channel 33	
	Channel 04		Channel 33	
	Channel 04		Channel 04	
	Channel 04		Channel 04	
	Channel 12		Channel 12	
	Channel 12		Channel 12	
	0		Channel 12	

Table 3.21: Example of the optimized compressed raw data sub-fragment. Three channels are packed in the 6 bytes format and two in the 10 bytes format. The extra info field in the header (0xB) indicates the Fragment 1 version used (MSB = '1') and the number of channels in the 6 bytes format (3).

Reconstruction Sub-Fragment

The reconstruction sub-fragment includes the result of the Optimal Filtering algorithm in a 32-bit word for each channel. The ROD can be configured to additionally compute the total sum of the energy and some projections for a superdrawer and the results are packed in three 32-bit words in the reconstruction sub-fragment. Thus, the total size of the sub-fragment including the header would be 54 or 51 32-bit words depending on whether the energy sum algorithm is requested or not. The first part of the sub-fragment includes the 48 words for the reconstruction result ordered with the channel index. The reconstruction word includes the gain, energy, time, HLT bad flag bit and the quality factor of the reconstruction (Table 3.23). The packing format of the reconstruction result is equal for the different flavors and

31	**...**			**24**	**23**	**...**	**16**	**15**	**...**	**0**
UU	PP	S	A	II	Sub-Frag Type: 0x04			Module ID		

Note:
U : Calibration Units. 0 - ADC ; 1 - pC ; 2 - Cesium pC ; 3 - MeV.
P : Pulse Shape. 0 -Physics ; 1 - Laser ; 2 - Charge Injection ; 3 - Simulated.
S : Number of samples. 0 - 7 samples ; 1 - 9 samples.
A : OF Algorithm. 0 - OF1 (2 parameters) ; 1 - OF2 (3 parameters).
I : Iterations. 0 - No Iterative algorithm; [1,3] - Number of iterations.

Table 3.22: Header format for the reconstruction sub-fragment.

31	**30 ..... 17**	**16 ..... 5**	**4**	**3 ..... 0**
Gain	Energy	Time	Flag	QF

Table 3.23: Reconstruction word data format including the selected gain, energy, time, bad channels flag and Quality Factor (QF).

configurations of the Optimal Filtering algorithm implemented in the DSP. These parameters are included in the extra info field in the header (Table 3.22). It includes the Optimal Filtering algorithm used (OF1 or OF2) as well as the units, number of samples and if the iterative method was used. The basis of the different methods is explained in Chapter 6.

The information included in the Optimal Filtering reconstruction result (Table 3.23) is encoded inside the DSP to take advantage of the maximum possible range and precision. This information can be decoded using Equations from (3.2) to (3.4).

$$Energy = \frac{A + offset}{Scale} \tag{3.2}$$

Where the scale depends on the calibration units used and the offset on the gain (Table 3.24).

$$Time = \frac{T - 2048}{16} \tag{3.3}$$

Where the precision of the reconstructed time is 0.0625 ns and the covered range is [-64,64] ns.

Units	Gain	Offset	Scale	E_{min}	E_{max}	Precision
ADC	LG	512	16	-32	2016	0.0625
	HG	2048	16	-128	1920	0.0625
pC	LG	512	32	-16	1008	0.03125
	HG	2048	2048	-1	15	0.00048828125
Cs pC	LG	512	32	-16	1008	0.03125
	HG	2048	2048	-1	15	0.00048828125
MeV	LG	512	0.03125	-16384	1032160	32
	HG	2048	2	-1024	15360	0.5

Table 3.24: Reconstruction word data format.

$$Quality = QF * Scale \tag{3.4}$$

Where the Scale for the Quality Factor (QF) depends on the Optimal Filtering method used: 32 for iterative method; 128 for non-iterative method.

The HLT bad flag bit is used to mark known bad channels that should not be considered by the HLT. The status of each channel is retrieved from a database and copied into the DSP memory at configuration time.

Three additional words are included at the end of the reconstruction fragment if the Level 2 E_T^{miss} fragment is selected in the TDAQ Tile Globals panel of the DSP configuration. These three 32-bit words contain the total sum of the energy, the transverse and the z-axis projection. In all the three cases the precision of the result is 0.5 MeV corresponding to the precision of the Optimal Filtering energy in HG. Table 3.25 shows the structure of the reconstruction fragment including the total energy sum fragment.

Data Quality Sub-Fragment

The DQ fragment is built by the InputFPGA and it contains the result of several digital checks that are used to detect errors in the transmission of the data. The size of this sub-fragment including the header is 9 32-bit words and the digital checks includes the status of the transmission with a DMU block granularity (3 channels). This fragment is used offline to flag group of 3 channels if any error is detected at ROD level. The received fragment from the

Byte	**3**	**2**	**1**	**0**
Start of header marker	0xFF1234FF			
Sub-Fragment Size	54			
Source ID	OF Info	0x04	Module ID	
Reco Data	Reco word Chan01			
	Reco word Chan02			
	.....			
	Reco word Chan47			
	Reco word Chan48			
SumE	Total E_T Sum			
	Total E_Z Sum			
	Total E Sum			

Table 3.25: Example of the reconstruction sub-fragment including the total energy sum words.

Byte	**3**	**2**	**1**	**0**
Start of header marker	0xFF1234FF			
Sub-Fragment Size	9			
Source ID	0	0x0A	Module ID	
DATA	**BCID**[27..16]		Global CRC	
	Memory Parity		**BCID flag**	
	Double Strobe		Single Strobe	
	Header Parity		Header bit 31	
	Samples Parity		Samples bit 31	
	ROD *DMU Chip Mask*		FE *DMU Chip Mask*	

Table 3.26: Data format of the ROD output Data Quality sub-fragment. The DSP adds the TTC BCID and the second bit of the BCID flag word with the result of the comparison between the second DMU BCID and the TTC BCID.

Input FPGA (Table 3.9) is modified by the DSP which adds the BCID used to synchronize the event and the result of the comparison between the second DMU and the TTC BCIDs. This allows to detect any eventual misalignment between the header and trailer DSPs.

3.8.5 Calibration Fragments Data Format

In addition to the data elements included in the bytestream by the ROD there are some calibration fragments which contain the front-end calibration information. The CIS parameters fragment is not written to the bytestream through the standard mechanism but it is added

Word	Content
1	Mode (0=normal, 1=calibration)
2	Samples (1-16)
3	Pipeline (0-255)
4	I3delay (0-255) between 3in1 charge and L1A generation)
5	Event (event number in 3in1 phase loop)
6	Phase (0-255)
7	DAC (3in1 dac setting, 0-1023)
8	CAP (3in1 capacitor, 5 or 100)
9	Card fired in CIS run
10	Reserved
11	Seconds from 1970
12	Microseconds
13	Run type
14	Reserved
15	Reserved
16	Event number

Table 3.27: CIS parameters raw data.

through an RCD data output to a ethernet readout ROS and then to the Event Builder. Table 3.27 describes the information included in this sub-fragment.

A special fragment is also used for Laser calibration . The Laser fragment (Table 3.28) is written to the bytestream by a dedicated Laser ROD.

Pedestals contain pedestals of Diodes 1-4 and PMTs 1-2. Alpha parameters and Pedestal for alpha parameters contain parameters of Diodes 1-4. Environment status contains Laser diode and box temperatures, also Laser box humidity and gas flow. Table 3.29 shows the definition of PLC status bits.

3.9 ROD Operation Modes and Bandwidth Limitations

The ROD is configured to provide different data elements depending on the type of run. The main constraint that does not permit to transmit complete raw data and reconstruction fragments is the ROD output data bandwidth. As seen in Figure 3.18 the output data bandwidth of the ROD is half of the input. This output bandwidth of a single ROL is 1.28 Gbps.

1	Laser Counter	`0xnnnnnnnn`	Number of Laser Trigger Type
2	SLAMA data	`0x20Cfaaaa`	`f` - Filter wheel, `a` - Requested laser pulse amplitude
3	SLAMA data	`0x21dddaaa`	`ddd` - Delay, `a` - Measured laser pulse amplitude
4	SLAMA data	`0x2200tttt`	`t` - TDC1 data
5	SLAMA data	`0x2300tttt`	`t` - TDC2 data
6	ADC data	`0x44aaabbb`	`a` - Channel 1 (Diode 2), `b` - Channel 0 (Diode 1)
7	ADC data	`0x45aaabbb`	`a` - Channel 3 (Diode 4), `b` - Channel 2 (Diode 3)
8	ADC data	`0x46aaabbb`	`a` - Channel 5 (PMT 2), `b` - Channel 4 (PMT 1)
9	ADC data	`0x47aaabbb`	`a` - Channel 7 (Spare), `b` - Channel 6 (Injected charge)
10	Diode 1 pedestal	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
11	Diode 2 pedestal	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
12	Diode 3 pedestal	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
13	Diode 4 pedestal	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
14	PMT 1 pedestal	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
15	PMT 2 pedestal	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
16	Pedestal timestamp	`0xssssssss`	`s` - Unix timestamp
17	Diode 1 alpha param	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
18	Diode 2 alpha param	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
19	Diode 3 alpha param	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
20	Diode 4 alpha param	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
21	Alpha param time	`0xssssssss`	`s` - Unix timestamp
22	Diode 1 ped for alpha	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
23	Diode 2 ped for alpha	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
24	Diode 3 ped for alpha	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
25	Diode 4 ped for alpha	`0xmmmmrrrr`	`m` - Mean * 10, `r` - RMS * 100
26	Ped for alpha	`0xssssssss`	`s` - Unix timestamp
27	Laser temperature	`0xsssssttt`	`s` - Sec since SOR, `t` - Measure * 10
28	Photodiodes temp.	`0xsssssttt`	`s` - Sec since SOR, `t` - Measure * 10
29	Photodiodes humidity	`0xsssssttt`	`s` - Sec since SOR, `t` - Measure * 10
30	Photodiodes gas flow	`0xsssssttt`	`s` - Sec since SOR, `t` - Measure * 10
31	Global status	`0xsssssttt`	`s` - Sec since SOR, `t` - Status bits

Table 3.28: Laser fragment.

The output data bandwidth used depends on the input event rate (Level 1 Trigger rate) and the size of the ROD output fragments (Equation 3.5). If the output link is saturated the ROD has to stop processing data and transmits the busy signal to the CTP to hold the generation of L1A signal and therefore inducing undesired dead-time.

11	10	9	8	7	6	5	4	3	2	1	0
ER	0	AL	IL	SC	SO	H2	H1	LV	A2	A1	A0

Note:
ER. Error status in the current measurement. 0-No error; 1-Error.
AL. Alarm status.
IL. Interlock open.
SC. Shutter closed (cannot be set together with SO or E should be 1).
SO. Shutter open (cannot be set together with SC or E should be 1).
HV. High Voltage 1,2.
LV. Low voltage.
A: Alpha 0,1,2.

Table 3.29: Laser sub-fragment status bits definition.

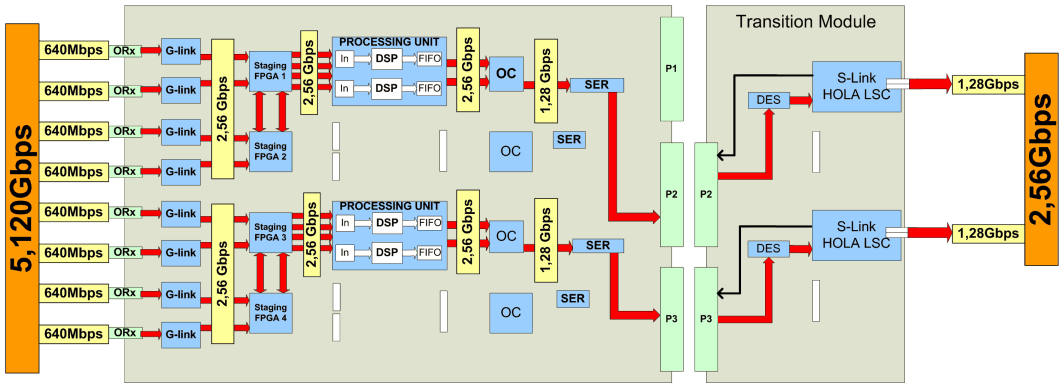


Figure 3.18: Data bandwidth limitations in the ROD data flow.

$$MaximumROLSize[words] = \frac{1.28Gb}{32} \frac{1}{Level1rate[Hz]} \tag{3.5}$$

Physics Data Format

According to Equation 3.5 the Level 1 Trigger rate establishes the maximum ROD fragment size that can be sent by the ROD without exceeding the limitations. At the nominal Level 1 Trigger rate of 100 kHz the ROD is capable of transmitting events through each ROL (4 super drawers) of 1600 kBytes (400 32-bit words) without introducing dead-time. As the Level1 trigger rate decreases, the size of the output fragment can be increased. During ATLAS

ROD Fragment Header
Reconstruction SD1
Compressed Raw SD1
Data Quality SD1
Reconstruction SD2
Compressed Raw SD2
Data Quality SD2
Reconstruction SD3
Compressed Raw SD3
Data Quality SD3
Reconstruction SD4
Compressed Raw SD4
Data Quality SD4
Status words
ROD Fragment Trailer

Table 3.30: Structure of the ROD output fragment for a physics run.

commissioning cosmics data taking and early LHC proton-proton collisions with very low peak luminosity the Level 1 Trigger rate was below 10 kHz allowing larger ROD output event sizes. During this period the complete raw data was transmitted together with the reconstruction fragment. This operation mode was used to validate the online reconstruction algorithms using the offline reconstruction as the reference.

With the increase of the luminosity and the associated increase of the Level 1 Trigger rates the complete raw data sub-fragment was replaced by the compressed raw data. Table 3.30 shows the structure of the present output ROD fragment for a physics run.

Calibration Data Format

There are two different data format for Calibration runs depending on the number of gains transmitted from the front-end. The online reconstruction function is not implemented in the DSP for bi-gain runs (Pedestals and CIS scan) and therefore only the raw data is transmitted in the output fragment (Table 3.31a). On the other hand, for auto-gain calibration runs (Laser and CIS mono) both the online reconstruction and the complete raw data are transmitted (Table 3.31b). The CIS ramp is a special calibration run used to validate the DSP Operation

mode. Therefore, in this case the physics configuration is used for the ROD and the output fragment includes the compressed raw data together with the reconstruction sub-fragment as shown for Physics runs (Table 3.30).

ROD Fragment Header
Raw Data bi-gain SD1
Raw Data bi-gain SD2
Raw Data bi-gain SD3
Raw Data bi-gain SD4
Status words
ROD Fragment Trailer

ROD Fragment Header
Reconstruction SD1
Raw Data SD1
Data Quality SD1
Reconstruction SD2
Raw Data SD2
Data Quality SD2
Reconstruction SD3
Raw Data SD3
Data Quality SD3
Reconstruction SD4
Raw Data SD4
Data Quality SD4
Status words
ROD Fragment Trailer

Table 3.31: Structure of the ROD output fragment for a) bi-gain (left) and b) auto-gain (right) calibration runs.

Chapter 4

Optical Multiplexer Board 9U

4.1 Introduction

As we have seen in the previous Chapters, upon the reception of the L1A signal the PMT samples are transferred from the eight digitizers to the Interface Board. Then, the samples are assembled, formatted and transmitted from the Interface Board to the back-end electronics through optical fiber links using the G-Link protocol. The Interface Board is also responsible of receiving the TTC information through an optical link from the back-end system, which is converted into an electrical signal and distributed to the eight digitizers of the superdrawer.

After a radiation tolerance test program the TileCal collaboration decided to use a redundant readout based on commercial components. Both, Single Event Error (SEE) and Total Ionizing Dose (TID) tests showed a small impact on the performance attributed to the expected radiation levels in the TileCal Interface Board [19]. The results showed that at the simulated radiation levels no increase in the transient of the supply current was observed during the TID tests [27]. On the other hand, during the SEE tests three types of non-destructive errors were observed; transient errors in the data stream, latch-up errors which produced an increase in the supply current and permanent data errors in the Altera FPGA. The transient increase in the supply current does not produce an observed impact in the performance of the system. The permanent data errors in the FPGA were recovered only after a complete

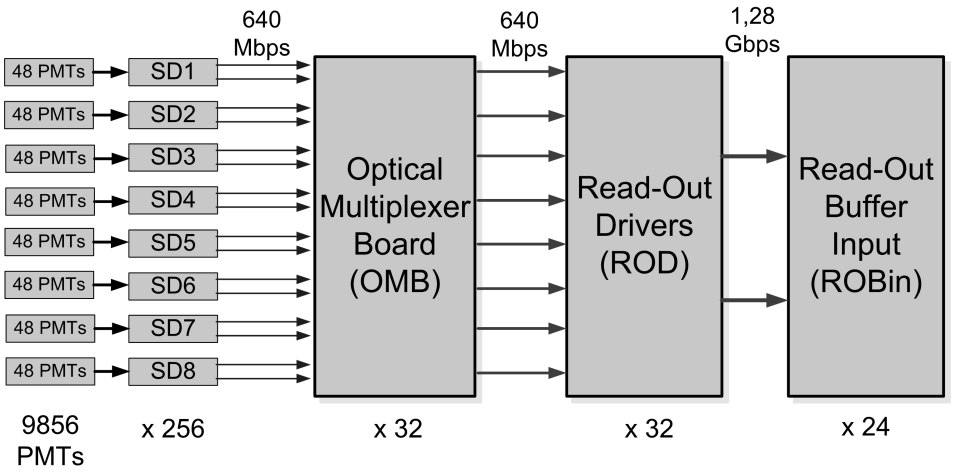


Figure 4.1: Sketch of the TileCal redundant readout logic. Each superdrawer transmits the data redundantly to the back-end. The OMB selects the link free of errors which is transmitted to the ROD for further processing. The reconstructed data is finally sent to the ROBin cards.

reset of the device which can be implemented as a power cycle of the front-end electronics. The tests estimated that a maximum of 3.7 power resets per week would be needed in the 256 TileCal superdrawers based on the simulated hadron fluence. Finally, a non-negligible number of transient errors or Single Event Upsets (SEU) in the data streams was observed. The effect of these transient errors could be significantly reduced by introducing a redundant readout circuit in the Interface Board. The basic idea is to duplicate the logic in the Interface Board and transit the data to the back-end system via two different optical links. The data from the digitizer boards (which are based on radiation hard components) is duplicated in the input of the interface board and two parallel data paths including two Altera FPGAs routes the data to the two optical transmitters. To detect transient errors a CRC is computed for each transmitted data packet and the result is appended at the the end of the data frame.

The main function of the OMB [37] is to receive the redundant data packets from the Interface Board, to check the CRC and decide in real time which of the two data packets is transmitted to the ROD (Figure 4.1). In absence of the OMB only one of the two redundant links is connected to the ROD, which was the operating mode during the first three years of ATLAS operation, with a negligible number of CRC errors observed.

Moreover, the OMB can be used to emulate the data produced in the front-end electronics. This injection mode has been used in the ATLAS electronics cavern when the detector was not available to inject data to the ROD system for calibration and tests purposes. It has been extensively used in the ROD tests-bench to qualify the ROD firmware developments before the installation in the real system and specially to validate and study the performance of the DSP reconstruction algorithms. The OMB can be connected to the TTC network allowing the emulated data to include actual TTC information, which can be used to test the synchronization algorithm of the ROD.

This Chapter describes the OMB module including the main components, operating modes and the monitoring and control software. The production and qualification tests of the OMB modules are presented in the next Chapter. The author was responsible for the design of the circuit, the development of the firmware and software and finally the production and qualification tests of the OMB project.

4.2 OMB Motherboard

The OMB motherboard uses a VME64X 9U standard format (Figure 4.2) [37]. It is able to receive redundant data from eight front-end superdrawers which corresponds to the number of ROD input links. Therefore a total of 32 OMBs, one per ROD, are needed to readout the complete TileCal detector. The eight RODs of each partition are placed in the even slots of the ROD crate. In principle, each OMB would be plugged in the neighboring odd slot of the associated ROD in order to simplify the connections (Figure 4.3).

The main components of the OMB are:

Optical connectors The optical connectors perform the conversion between optical and electrical signals. The asymmetry between the number of inputs and outputs needed in the OMB (16:8) was crucial in the selection of the optical connectors. The typical optical transceivers provide full duplex connections with one input and one output per connector. Although very rare in the fiber connection market an alternative solution was found. The Stratos Ltd. dual optical receiver (M2R-25-4-1-TL) and transmitter (M2T-25-4-1-L) [38] have

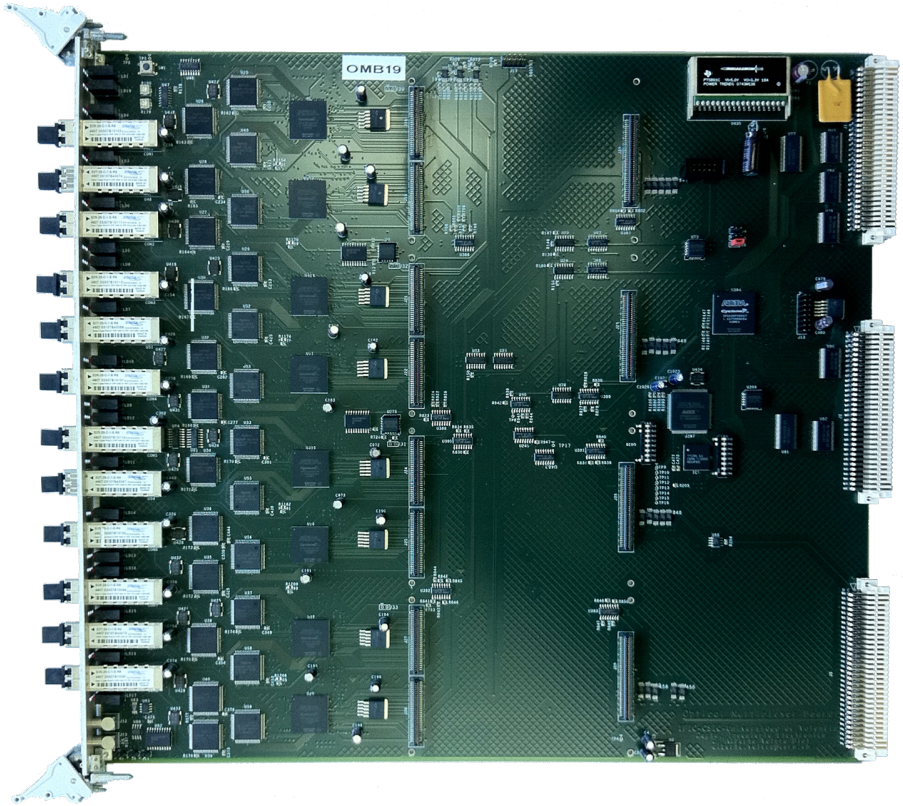


Figure 4.2: Picture of the OMB motherboard.

been chosen to optimize the space in the board since they provide uni-directional receivers and transmitters. The optical connectors are operated at a effective data rate of 640 Mbps (16-bit words at 40 MHz) to fulfill the specifications of the TileCal front-end links.

G-Link chips The electrical serial lines coming from the dual receivers are deserialized in the Agilent HDMP-1034 G-Link chips [30] ('D' in Figure 4.4). The chip is controlled from the CRC FPGA and a local oscillator provides a 40 MHz to each single chip. The deserialized 16-bit bus, control words and the 40 MHz clock are transmitted from the HDMP-1034 chip to the CRC FPGA. The serializer Agilent HDMP-1032 G-Link chips [30] ('S' in Figure 4.4) performs the opposite function for the output links of the OMB. In this case, the 16-bit data bus, the 40 MHz clock and control signals are transmitted from the CRC FPGA to the

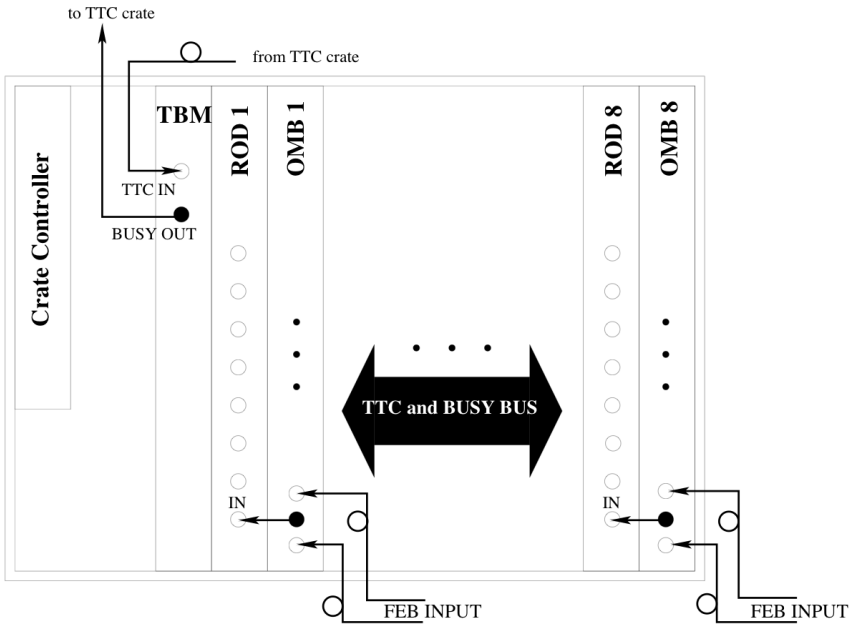


Figure 4.3: Sketch of a ROD crate with the OMB modules.

serializer chips. The output serial lines are connected to the optical transmitters. In total there are 16 deserializer chips connected to the 8 optical dual receivers and 8 serializer chips connected to the 4 dual optical transmitters. The G-Link chips are controlled and configured from the CRC FPGAs.

CRC FPGA The main processing device of the OMB are the eight CRC FPGAs which are Altera CYCLONE EP1C12 devices [39]. Each CRC FPGA can receive the data from two superdrawers through a dual receiver connector, and can transmit the data through one optical link in a dual transmitter connector. The reception and transmission of data is performed through the G-Link deserializer and serializer chips. Moreover, the CRC FPGA configures these chips and monitors the status of the G-Link chips. Two EPCs memories are used to configure the four upper and four lower CRC FPGAs at startup. Moreover, each single CRC FPGA can be configured through the JTAG chain. The VME access is implemented in a daisy chain serial bus connecting the four upper and four lower CRC FPGAs in two different

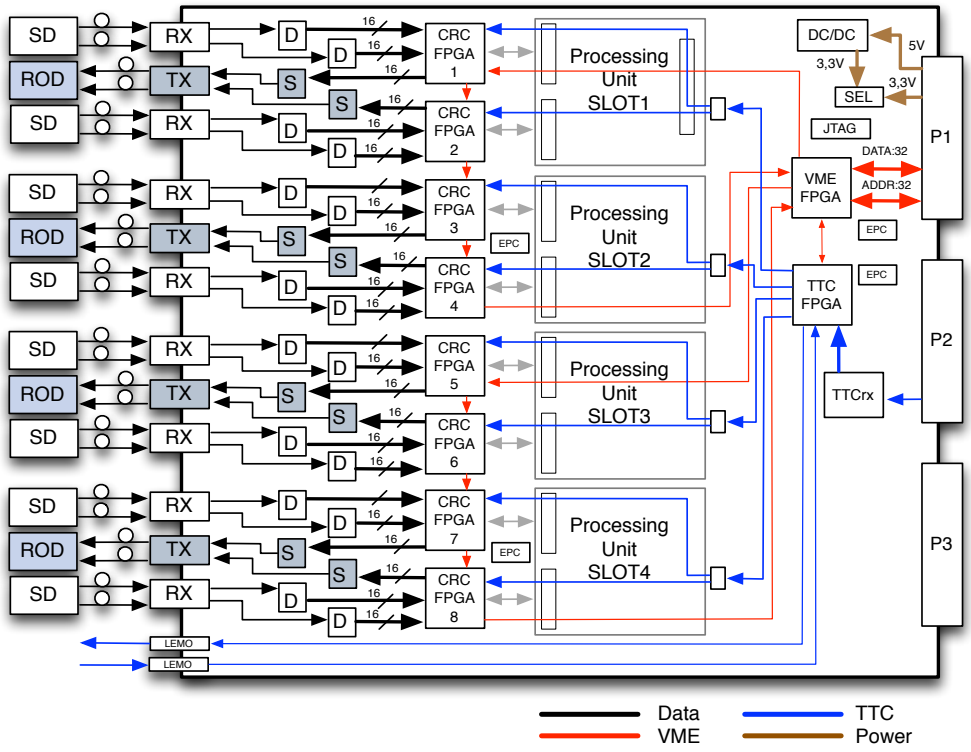


Figure 4.4: Block diagram of a OMB module with the different data paths.

busses (Figure 4.4). Hardware switches are used to provide different addresses to each device in the daisy chain, which is used to map the internal registers.

PMC Processing Units slots Each OMB includes four PMC slots which are pin-to-pin compatible with ROD PU daughter boards. They might be used to upgrade the processing power of the OMB. Each slot has bi-directional communication with two CRC FPGAs to receive and transmit data from up to four superdrawers.

VME FPGA There is one VME FPGA which is an Altera Cyclone EP1C20 device [39]. The VME FPGA is the interface between the OMB devices and the VME bus. The geographical address of the board is obtained during the initialization of the board and depends on

the slot where the board is plugged into the VME crate. The address of each register inside the board is composed by the geographical address, common for all the registers in a board, and a local address mapped in the VME FPGA. On one hand, the communication with the VME bus is performed through 32-bit address and data bi-directional busses to implement read and write cycles. On the other hand, the communication with the devices in the board is performed through a specific serial protocol. A dedicated bus is used to communicate with the TTC FPGA whereas two daisy chained serial busses provides the communication with the CRC FPGAs (Figure 4.4). A complete description of the internal functions of the VME FPGA can be found in the Section 4.2.2.

TTC System The TBM receives the TTC information from the TTC crate through an optical input. The signals are converted to electrical and distributed through the VME P3 backplane to all ROD or OMB boards plugged into the crate through point-to-point serial connections. The serial signals are decoded in the TTCrx which provides the information to the TTC FPGA. The signals decoded and transmitted to the TTC FPGA are the L1A, ECR, BCR, the 40 MHz Bunch Crossing clock and addressed commands which are used to transmit TTypes. The TTCrx gets its address on startup from a configurable hardware switch (Figure 4.5). The TTC FPGA receive these signals from the TTCrx and internally builds and stores the TTC information which is broadcast to all the CRC FPGAs. The same output information is transmitted through four different ports, each of them duplicated to transmit this information to the eight CRC FPGAs. The internal operation of the TTC FPGA is described in the Section 4.2.3.

Configuration and memories The programmable devices and memories are all connected to a JTAG chain which can be controlled either from the JTAG connector or from the VME FPGA. The JTAG connector allows to configure all the programmable devices in the chain including the FPGAs volatile configuration and it is used for debugging and tests purposes in the lab. The remote access to the JTAG chain through the VME FPGA allows only the programing of EPC memories and a power cycle of the board is needed to transfer the configuration to the FPGAs. This configuration mode is used to remotely update the OMB and ROD firmware in USA15. Figure 4.6 shows the connection of devices within the JTAG

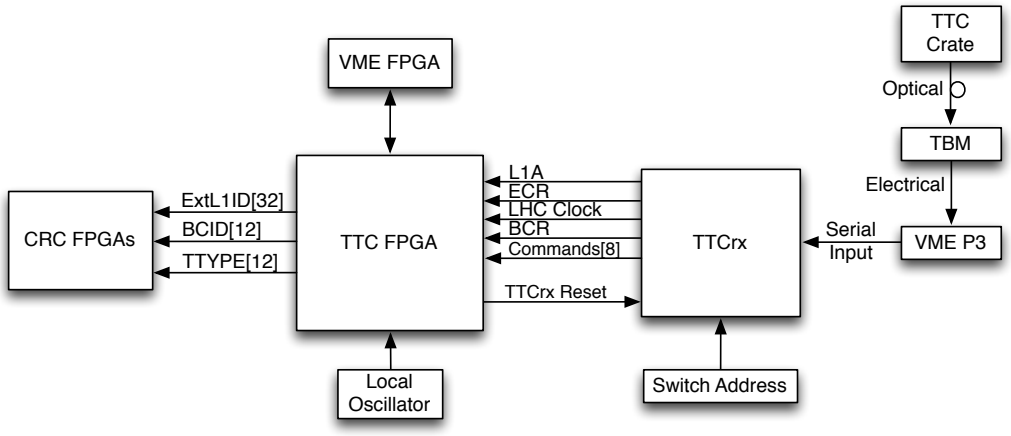


Figure 4.5: Distribution of TTC information in the OMB.

chain. The black line shows the connections of JTAG signals between the components while the dashed arrows show the configuration of FPGAs from EPCs at every startup of the OMB module. The TTCrx ASIC cannot be programmed but it is included in the JTAG chain to verify and initialize the device through JTAG commands.

The OMB motherboard provides three different data paths which are shown in Figure 4.4.

- The data-flow (black line in Figure 4.4) corresponds to the input data from the superdrawers and the output data to the ROD. The redundant data from a superdrawer is received through a dual optical receiver, deserialized in a G-Link chip and transmitted to a CRC FPGA. The CRC FPGA performs the CRC check and the selected event is serialized and transmitted to the ROD through one of the two output links of a dual transmitter. In the injection operating mode the data is generated in the CRC FPGA, serialized and transmitted to the ROD.

- The VME data path (red lines in Figure 4.4) is used for configuration and monitoring purposes. The interface between the OMB devices and the VME bus is implemented in the VME FPGA. It defines the internal registers mapping whereas the absolute VME address is completed with the geographical address which depends on the slot the board is plugged in. The read and write access between the crate controller and the

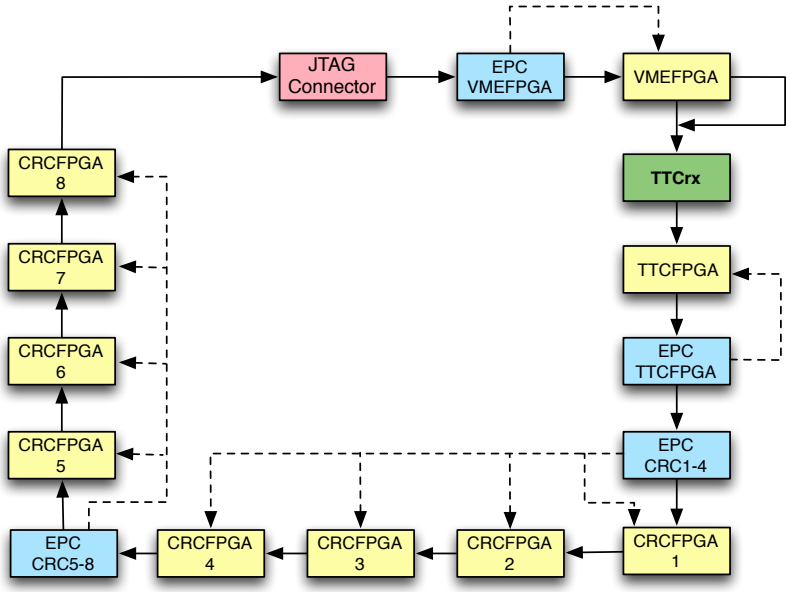


Figure 4.6: Block diagram of the OMB JTAG chain with FPGAs and memories.

VME FPGA are performed through two 32-bit busses for address and data. Internally, the VME FPGA communicates with a dedicated serial line with the TTC FPGA and through two daisy chained serial lines for each group of four CRC FPGAs.

- The TTC path (blue lines in Figure 4.4) is used to receive the TTC information through the VME backplane which is broadcast to each CRC FPGA.

4.2.1 The CRC FPGA Firmware

The CRC FPGA is the main processing component of the OMB board. The eight CRC FPGAs in each OMB use the same firmware code. Figure 4.7 shows the main functional blocks and data-flow inside the FPGA. The three dashed squares correspond to the three main functional blocks. These three modules are separated logic regions inside the FPGA which can be compiled and configured independently. The Altera Cyclone EP1C12 device has 12000 logic elements and 29 kBytes of dedicated memory blocks. The final version of the firmware occupies the 85% of the logic elements and 90% of the total memory.

CRC Checking block The CRC Checking block uses 2900 logic cells and 7 kBytes of dedicated memory blocks. The data events from each of the two input links are stored in three FIFO memories. Upon the reception of the event header the module starts the storage of the event in one of the FIFOs and simultaneously the Global CRC and the CRC of each DMU data block are computed. When the trailer of the event is received, the computed and received CRC words are compared and the result is transmitted to the Decision Logic block. The event with correct CRC is retrieved from the corresponding memory and transmitted to the output, while the other memory is cleared. In case of a CRC error in both links, the event with less DMU CRC errors is transmitted. In case of the same number of DMU CRC errors, one of the two events is transmitted randomly. The Decision Logic block allows the manual selection of a given input link for all the events.

Injection block The injection block can generate pseudo-data to be injected into the RODs, with three different types of data:

- The Counter mode generates events where all the data words inside an event are equal. The value of the word is incremented event by event.
- The Pulse Generator mode emulates real PMT pulses for a configurable energy and phase. This operation mode is used to validate the amplitude and phase reconstruction in the DSPs of the ROD.
- The Memory mode transmits events which are previously downloaded through the VME bus into an internal memory.

There are also two different trigger modes. If the TTC synchronization is selected, the events are generated upon the reception of a L1A signal and they include the associated BCID. If the TTC synchronization is not selected, the generation of events is triggered either internally or from the VME bus. For each injected event the computed CRC is included in the event.

In the so-called Error Injection mode, the CRC included in the data event is intentionally wrong in the data injected by the four even CRC FPGAs when the BCID associated is even, and it is wrong in the four odd CRC FPGAs when the BCID is odd. The index of the CRC FPGA are defined using a hardware switch which uniquely identifies each CRC FPGA from

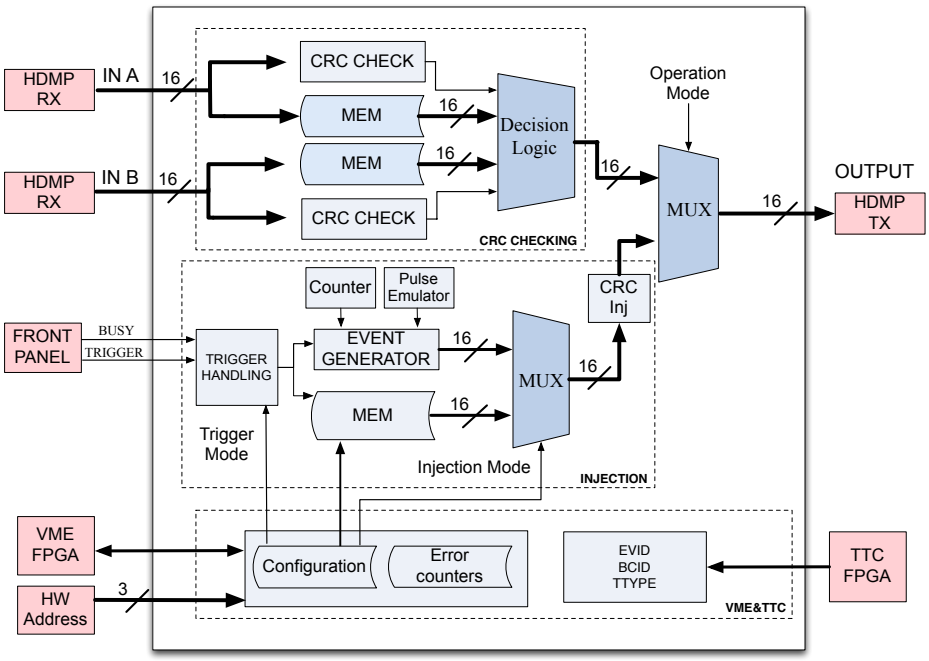


Figure 4.7: Block diagram of the CRC FPGA. The three dashed squares refers to separate logic regions inside the FPGA.

0 to 7. Therefore, in this operation mode half of the output links of the OMB always have a wrong CRC while the other four are correct. This mode is used to verify the correct selection of links by injecting data to another OMB module. A complete description of the various operating modes can be found in Section 4.3.

VME and TTC interface block The VME and TTC interface block uses more than 5000 logic elements. It does not use dedicated memory blocks since the configuration and error counters are stored in the registers of the logic elements. This module provides communication with the VME FPGA which is used to write the configuration and read the status and values from the internal registers. It also manages the download of data events into the memory for the Memory injection mode. The TTC part receives the BCID, L1ID and TTYPE from the TTC FPGA. The information of the last received TTC event can be retrieved through the VME interface.

4.2.2 The VME FPGA Firmware

The VME FPGA implements the interface between the VME bus and the OMB. The final version of the firmware uses 2000 logic elements in the FPGA with reduced use of memory blocks since the error counter registers are physically located in each CRC FPGA. The communication with the VME bus is managed by the VME crate controller which is the master of the bus whereas the OMBs act as slave modules. Therefore, the VMEbus Inteface block (Figure 4.8) cannot start neither a write nor a read cycle, which is always requested by the crate controller. During an access cycle the address of the accessed register is provided by the controller. If the access is a read cycle, the VME FPGA configures the driver to transmit the value of the requested register from the OMB to the VME connector. In a write cycle the driver is configured in the opposite direction and the data is transferred from the VME connector to the VME FPGA. If the access correspond to the geographical address of the board, the OMB Registers Mapping block is activated to obtain the final placement of the register. If the register is placed in the VME FPGA then it is accessed in the VME FPGA Registers module. If it corresponds to an external device then the communication protocol with the appropriate device is started. The complete address of any register is composed by 3 fields:

ADDRESS[31..0] = GEOG[31 .. 12] + DEVICE[11 .. 7] + REGISTER[6 .. 0] where the geographical address depends on the slot where the module is plugged in; the internal device address is defined in Table 4.1; and the internal register depends on the type of device (CRC FPGA, VME FPGA or TTC FPGA).

The communication with the CRC FPGAs is performed through two daisy chained busses. The busses contain four lines for clock, frame, data and reset. The custom protocol is instantiated twice, one for the four upper and one for the four lower CRC FPGAs. The communication with the TTC FPGA is performed through a similar 4 lines bus but with point-to-point connection.

Device	Binary Address[11 .. 8]	Hex Address[11 .. 8]
CRC FPGA 1	0000	0x0
CRC FPGA 2	0001	0x1
CRC FPGA 3	0010	0x2
CRC FPGA 4	0011	0x3
CRC FPGA 5	0100	0x4
CRC FPGA 6	0101	0x5
CRC FPGA 7	0110	0x6
CRC FPGA 8	0111	0x7
VME FPGA	1000	0x8
TTC FPGA	1001	0x9

Table 4.1: OMB device address mapping.

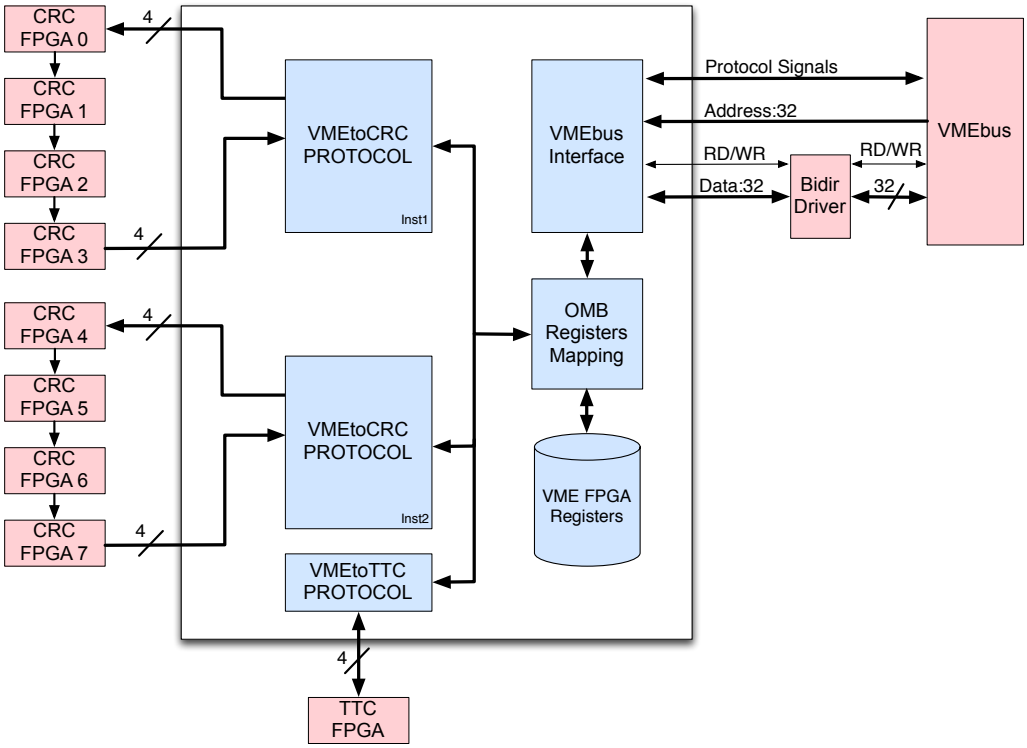


Figure 4.8: Block diagram and data flow of the VME FPGA firmware.

4.2.3 The TTC FPGA Firmware

The TTC FPGA uses the decoded TTC signals provided by the TTCrx to build and distribute the TTC information to all the CRC FPGAs. The ACEX EP1C30K device used provides 1120 logic elements and 3 kBytes of dedicate memory blocks. The final version of the firmware uses the 65% of logic elements and less than 10% of the memory.

Figure 4.9 shows the main functional blocks of the TTC FPGA. The *TTCtoVME* provides communication with the VMEbus to read and write in the internal TTC FPGA registers and to monitor and reset the TTCrx ASIC. The Clock Selection block determines which clock is used in the OMB. If the LHC clock distributed by the TTCrx is present, it is distributed to all the devices in the board. If the selector detects any problem or the LHC clock is not present, then a 40 MHz clock provided by a local oscillator is used. It is also possible to force the usage of the local oscillator clock. The selected clock is driven by several zero delay external clock buffers and distributed to all the CRC FPGAs.

However, the main function of the TTC FPGA is the acquisition and distribution of TTC information. It receives the LHC clock (Bunch Crossing clock), ECR, BCR, L1A and addressed commands from the TTCrx and generates the Extended L1ID, the BCID and the TType.

Upon the reception of a L1A signal, these three values are distributed to the eight CRC FPGAs through four 5-bit output busses. The 5-bit busses are composed by two pairs of data and frame signals. The first pair is used to sequentially transmit the ExtL1ID and the BCID whereas the second pair is used to transmit the TType. The additional line is used to reset the buffers in the CRC FPGAs from the TTC FPGA.

4.2.4 Printed Circuit Board and Power Distribution

The OMB motherboard is a 10 layer Printed Circuit Board (PCB) that optimizes the cross-section to minimize signal integrity problems. The circuit contains a total of 1320 components and 2337 connections. The main components are placed in the top layer with some termination resistors and decoupling capacitors in the bottom layers. The signal layers are kept between two power planes (Figure 4.10). The routing of two adjacent layers is done orthogonally. In

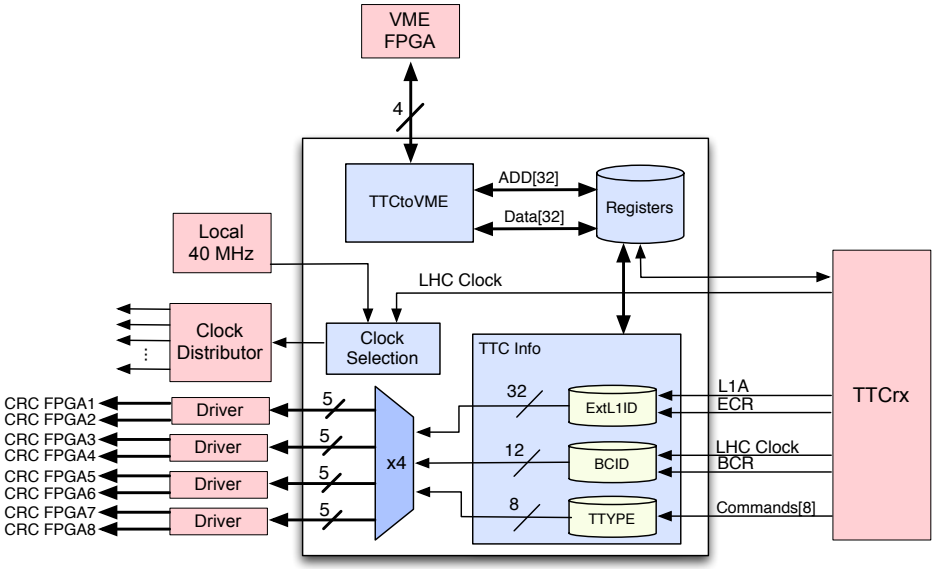


Figure 4.9: Block diagram and data flow of the TTC FPGA firmware.

addition to the 6 signals layers (top, bottom and four internal) there are two ground layers and two power layers.

The power distribution in the board is defined by the several different supply voltages used by the components. The main power supply is provided by the VME bus. There are two operating modes depending on whether the main 3.3 V is directly taken from the VME bus or if a DC/DC converter placed in the board is used to generate the main 3.3 V from the 5 V provided by the VME bus (Figure 4.4). The operation mode is selectable placing a fuse in the corresponding socket in the board. The two operating voltages can be used to balance the power consumption of 3.3 V and 5 V from the VME main power supply. The FPGAs use 3.3 V for the I/O and 1.5 V for the internal operations. The *Power1* plane is entirely used to distribute 3.3 V to most of the components in the board including the FPGAs. On the other hand, *Power2* layer is divided into several planes to distribute 1.5 V to the FPGAs. One plane placed below the 8 CRC FPGAs is supplied by eight 3.3 V to 1.5 V DC/DC converters mounted close to each FPGA. Two more power planes are placed below the VME and TTC FPGAs with the corresponding DC/DC converters. The remaining space

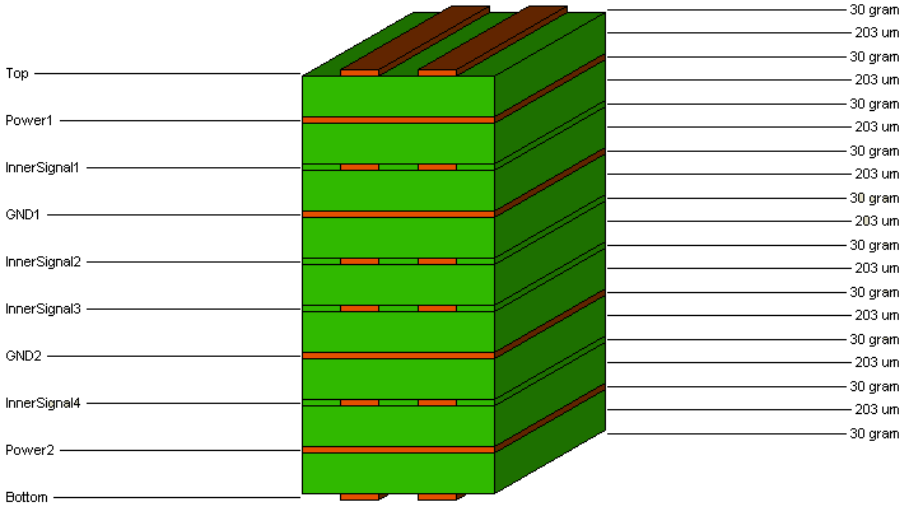


Figure 4.10: Stackup and dimmensions of the PCB. It consists of two layers for power, two for ground and six for signals.

on *Power2* layer is also used to distribute 3.3 V. The Transistor-Transistor Logic (TTL) to Nuclear Instrumentation Module (NIM) logic converters for the trigger inputs are supplied using the 12 V power supply lines in the VME backplane whereas the 5 V TTL devices are supplied using the same 12 V lines plus additional local regulators.

4.3 Operating Modes

The main operating mode of the OMB is the CRC checking although the injection mode has been extensively used to emulate the TileCal front-end electronics for ROD tests and firmware developments. The function of the CRC FPGA depends on the operating mode whereas the VME and TTC FPGAs are independent of the operating mode.

4.3.1 The LHC Mode

The OMB has been designed to detect and reduce the amount of events with corrupted data due to SEU in the front-end Interface Boards. In this operating mode, each CRC FPGA

receives the redundant data from one front-end superdrawers through two different inputs. For each input link a structure of three FIFO memories is used. When the trailer of an event is received, the link with correct CRC is transmitted to the ROD. If a new event is received during the transmission of a previous event it is stored in a different FIFO memory to avoid any memory conflict. In case of miss-synchronization of the two links, the decision logic might need some extra time until the events are completely received in the two links. The structure with three memories provides the buffering needed in case of miss-synchronization. This operation mode has been successfully tested in all the OMB produced above the ATLAS requirements which are a Level 1 trigger rate of 100 kHz with a complex deadtime of 8/350 (maximum of 8 L1A in 350 consecutive bunch crossings) and a simple deadtime of 4 (minimum difference between two consecutive L1As). The decision logic selects the first received event with correct CRC. If a Global CRC error is detected in the two redundant events the one with less number of DMU CRC errors is transmitted. If the same number of DMU CRC errors is also detected then the first event received is transmitted. These events with digital errors are discarded by the offline reconstruction software.

The statistics of detected errors are stored in internal registers in the CRC FPGAs and can be retrieved through the VME bus. The monitoring provides information about digital errors for all the events passing the Level 1 trigger. The information available includes, for each link, the number of Global CRC errors, DMU CRC and BCID errors and total parity errors. It also provides information about the number of received and transmitted events as well as the number of transmitted events with Global CRC errors (in the case of two redundant events with CRC error).

4.3.2 Injection Modes

The injection mode is used to emulate the injection of data events to the RODs. In this mode the input links are not used and the data generated in the CRC FPGAs are transmitted through the output links. Therefore, one OMB can inject data to one ROD module.

There are three different injection modes depending on the type of data generated in the OMB. In the three cases the injection of a data event should be triggered externally either through the front panel NIM connector or at the reception of a L1A signal in the TTC FPGA.

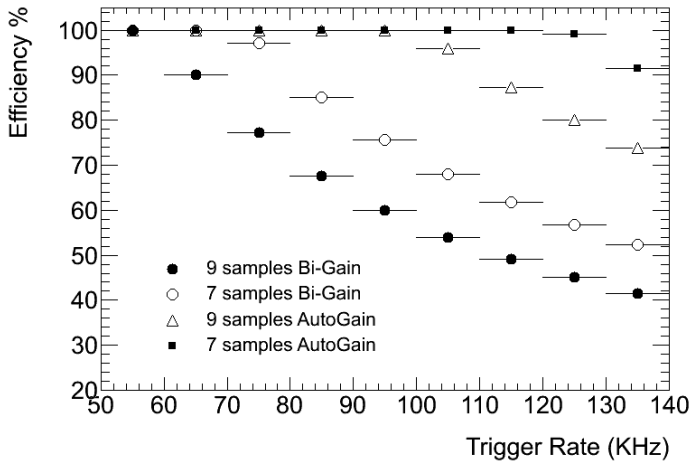


Figure 4.11: Injection efficiency as a function of the event size (defined by the number of samples and gains).

Counter Mode

In this mode the 16 DMU data blocks within an event includes a constant header (with correct BCID updated for each event) and trailer whereas the data words inside the event are equal. The value of the data words is incremented for each injected event. The Global CRC is computed for each event and appended at the end of the data. It is possible to configure the OMB to include a wrong Global CRC result in the event for half of the links in order to test the link selection injecting data to another OMB. The maximum injection rate is limited by the bandwidth of the output link. Therefore it depends on the event size which is defined by the number of samples and gains selected. The OMB is able to inject events with the ATLAS physics operation mode (7 samples and one gain) above the nominal Level 1 Trigger rate of 100 kHz (Figure 4.11).

Pulse Generator

The Pulse Generator (or Inverse Optimal Filtering) mode also includes the correct BCID in the header and the Global CRC at the end of the event. The important feature of this mode is that it produces the data samples for each of the three channels in a DMU data block for a

given amplitude, phase and pedestal. The data of the 16 DMU block are equal. The TileCal normalized pulse shape samples in steps of 1 ns are stored in three identical memories (one for each channel in the DMU block) in each CRC FPGA.

This operation mode is used to verify the performance of the Optimal Filtering algorithm in the ROD under ideal conditions. The maximum injection rate in this operation mode is limited by the output data bandwidth of the link (Figure 4.11) if the same samples are transmitted in all the events. If a different pulse configuration is required for each injected event, then the maximum trigger rate is limited by the VME bus and it is reduced to rates below 500 Hz.

Memory Mode

In the injection mode data previously copied in an internal memory inside the CRC FPGAs is transmitted with the reception of a trigger signal. The data event is not modified by the OMB and therefore this operation mode can not be used to synchronize the data and the TTC information at ROD level since the BCID of the events is constant. This operation mode can be used to reproduce the processing of real data in the detector in the laboratory test-bench using raw data files stored in ATLAS. Moreover, it is possible to build a data event using external software, copy it in the OMB memory and trigger the injection to the ROD. As we will see later there are some applications in the OMB software library to emulate different kind of PMT pulses.

The maximum trigger rate for this injection mode is limited by output data bandwidth if the same event is transmitted always. If the event is changed event by event then the limitation comes from the VME bus bandwidth and the maximum injection rate is reduced to few Hz.

4.4 VME Library and Software for OMB

The OMB software is a set of libraries with basic functions to read and write the OMB internal registers. It includes the corresponding header files, to define the address of all the registers, and a graphical user interface panel to configure and debug the OMBs. In addition, the

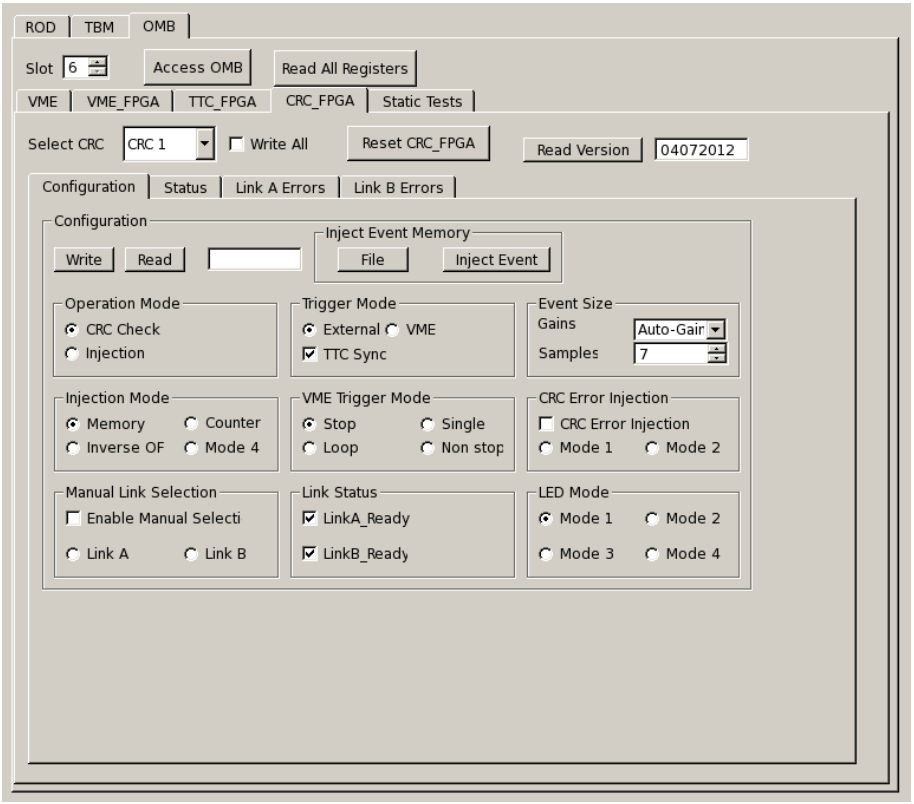


Figure 4.12: Snapshot of the OMB tab in the TileRODGui test software.

library includes some binary applications to download different type of events into the OMB for the Memory injection mode. The *OMB2RODInjector* application selects events for eight TileCal modules from an ATLAS raw data file. The events are copied into the eight CRC FPGAs and injected to the ROD for tests. The *OMBPulseinjector* generates the samples of PMT pulses for a given amplitude, phase and pedestal, builds and download the event with the correct format in the CRC FPGAs memories and finally injects the event to the ROD. This application is similar to the Pulse Generator injection mode of the OMB. However, the flexibility of the software allows the implementation of more sophisticated features, like the addition of pseudo-random noise to the samples (to emulate the real electronic noise) as well as the emulation of pile up pulses.

TileOMBGui

The *TileOMBGui* is the debug expert application to configure, test and monitor the OMB modules. It has been integrated in the more general *TileRODGui* application which includes also the access to the ROD and TBM modules. The OMB section of the software is divided into five tabs (Figure 4.12).

The VME tab allows the access to registers of any module plugged in the VME bus. There are three tabs for the VME, TTC and CRC FPGAs. In the later it is possible to select which of the eight CRC FPGAs is accessed. The CRC FPGA also includes different tabs for the configuration, status and to monitor the number of errors detected for each input link. Finally, the Static Test tab can be used to verify the accessibility to all the registers in an OMB board.

Chapter 5

Production, Commissioning and Operation of the Back-End System

5.1 Introduction

In the previous Chapters we have described in detail the readout electronics of the TileCal. The superdrawers acquire the signals from the PMTs which are digitized and transmitted to the back-end. Then, the RODs reconstruct the energy and time from the digital samples and the results are provided to the next trigger level. We have also seen how the detector is partitioned in four cylinders (two in the LB and two EB) sub-divided in 64 superdrawers each. Accordingly to that ,we have four so-called TTC partitions in the back-end system; two partitions are used to readout the LB and the other two to readout the EBs. Each TTC partition has a TTC crate and a ROD crate with eight RODs each. Thus, a total of 32 RODs are required to readout the TileCal detector plus 32 OMBs in the case of an increase in the number of SEU errors in the Interface Board. In this Chapter we describe first the production and qualification tests of the ROD and OMB boards. Following, the

installation and commissioning of the complete TileCal back-end system in the ATLAS cavern is presented. Finally, we show some performance results of the back-end system during the combined cosmics ray data taking of the commissioning period and the first three years of collisions in the LHC.

5.2 Production and Qualification Tests

A total of 32 ROD boards are required to readout the TileCal detector. In addition, 6 spare units were produced considering 10 years of ATLAS operation. Therefore, the production consisted on the fabrication of 38 ROD boards and the corresponding 38 TMs. In addition to that, 5 TBMs and 5 custom P3 VME back-planes (4 required plus 1 spare unit) were also produced.

The fabrication and assembly of all the modules were carried out by the University of Geneva together with the modules required by the LAr calorimeter. The boards were delivered to the IFIC-Valencia group during the spring of 2005. The first operation was to adapt the modules for TileCal requirements which consisted on minor hardware adjustments and in the reconfiguration of all the programable devices. The hardware adjustments are mainly related to the frequency of the G-Link protocol which is 80 MHz for LAr and 40 MHz for TileCal. It required a replacement of the oscillators used to feed the G-Link chips as well as some resistors used to select the correct frequency operation range of these chips. However, the main difference between the LAr and the TileCal ROD boards is in the firmware. TileCal uses a different firmware in the Staging, Input and TTC FPGAs as well as in the DSPs code. The production and qualification tests of the ROD boards were also used to validate the performance of the TileCal specific firmware. Moreover, the test-bench designed and installed to validate the ROD boards has been subsequently used to develop and evaluate firmware updates.

Once the boards were adapted, they had to pass a sequence of tests to become qualified and ready for installation. Each ROD was associated with two PUs and a TM. The whole system was tested together and this association was kept for the installation of the RODs in the ATLAS cavern. The results and incidences encountered during the tests were introduced

in a production database. In addition, the database includes the association of each ROD with the corresponding TM and PUs as well as the slot and partition where they are installed. It allows to correlate future problems with the production issues of a specific module.

In order to emulate in the laboratory the operation conditions, a complete custom test-bench was designed, which had to inject data events to the ROD as well as to store and to analyze the processed data coming from the RODs. The test protocol consisted in a sequence of tests with different injection rates and event sizes in order to simulate the conditions of real physics and calibration runs.

5.2.1 The ROD Production Test-Bench

The production test-bench was designed to simulate the operation conditions of the back-end electronics. It required the emulation of the detector data events as well as the TTC signals and it was implemented in two 9U VME crates, a NIM crate for trigger emulation and a ROS computer to store and data analysis (Figure 5.1). The NIM crate was used to generate trigger signals which were transmitted to the injection crate where the data events were generated. Finally, the ROD crate held the RODs under evaluation which processed the data events and transmitted the results to the ROS computer where the events were stored and analyzed. The boards used to emulate the propagation of TTC and busy signals as well as the SBC controllers were ATLAS standard boards. The ROD Injector and Multiplexer BOard (RIMBO) [40] and Optical Buffer (OB) [41] boards were used to emulate the TileCal front-end signal.

5.2.2 The Trigger Crate

The NIM trigger crate was used to generate L1A trigger signals with a commercial dual timer module. The rate was configured in the dual timer according to the requirements of each test level. In addition, the dual timer provided a veto input signal to stop the generation of pseudo Level 1 trigger signals in case of a busy status in the ROD crate (dashed black arrows in Figure 5.1). The L1A signals generated in the dual timer were transmitted to the Injection crate to trigger the generation of data (solid black arrows in Figure 5.1).

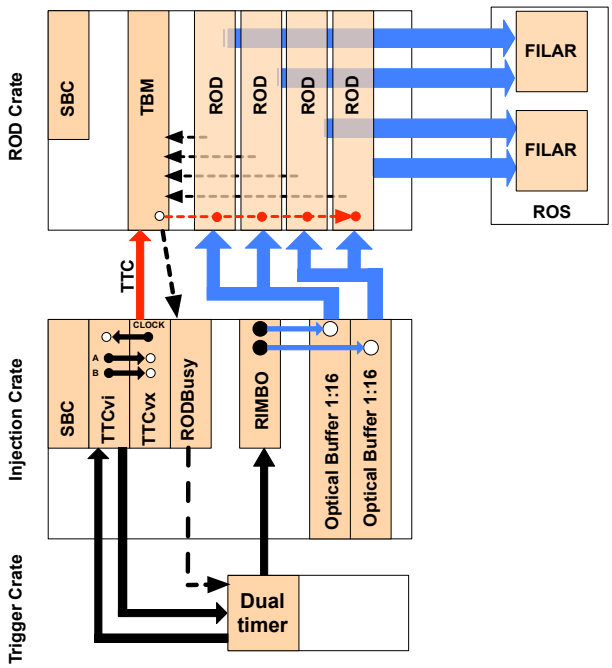


Figure 5.1: Sketch of the ROD production test-bench at IFIC-Valencia laboratory. Solid black arrows represent the generation of L1A signal. Dashed black arrows shows the propagation of busy signals. Red arrows show the transmission of the TTC information and blue arrows the flow of data events.

5.2.3 The Injection Crate

The injection crate included an SBC for configuration and monitoring purposes, a set of TTC ATLAS modules for the propagation of TTC and busy signals, the RIMBO (Figure 5.2) and two OBs. Essentially, upon the reception of a trigger signal the RIMBO generated a data event with the TileCal front-end data format. The two output links of the RIMBO were connected to two different OB modules. The OB is a 9U VME module specially designed to split each output of the RIMBO to up to 16 links. Each OB module has an optical input connector with a bandwidth of 640 Mbps and 16 optical outputs with a total bandwidth of 10.24 Gbps. Therefore, one RIMBO board and two OB modules can drive 4 complete RODs (half TileCal partition).

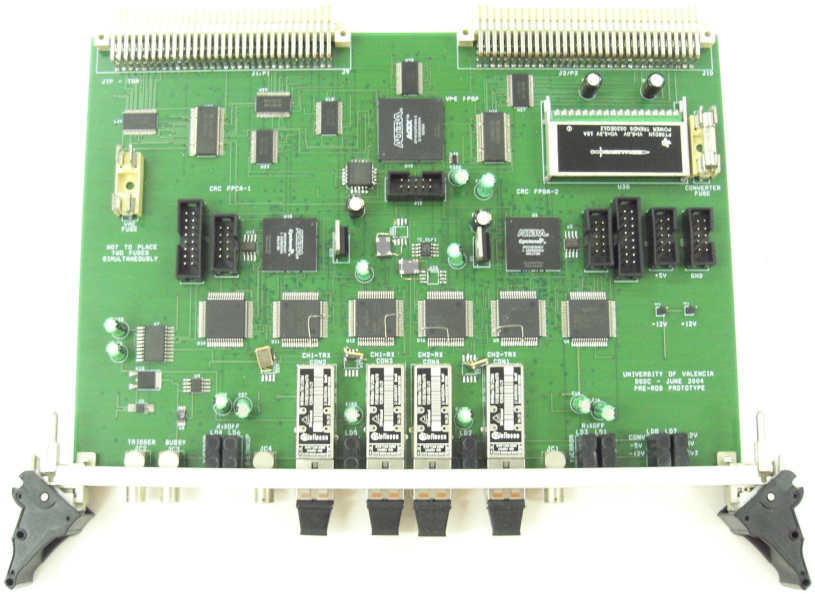


Figure 5.2: Picture of the components side of the RIMBO board.

The RIMBO Module

The RIMBO was designed by the IFIC-Valencia group with two main purposes. First, it emulates the flow of data coming from the front-end electronics to the RODs for the different data format types (physics and calibration). Second, it was used to evaluate the redundancy readout logic of TileCal. Essentially, the RIMBO board was used to study the technical viability of the OMB 9U module and to develop the firmware needed to check and transmit the correct data to the ROD in real time (Figure 5.3).

Therefore, the RIMBO includes two operation modes: the injection and the CRC checking modes although only the injection mode was used in the production test-bench. The injection of data to the ROD is driven by a trigger signal. The RIMBO has different trigger modes which are mainly used to adjust the injection trigger rate according to the qualification test protocol. High trigger rate are used for burn-in tests whereas single events are used to qualify the performance of the ROD reconstruction.

There are two different generation modes in the RIMBO: Counter and Memory modes.

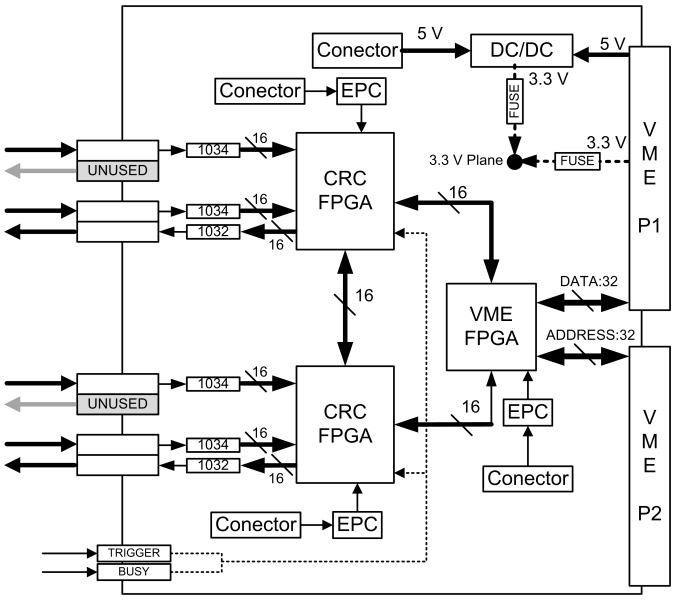


Figure 5.3: Sketch of the RIMBO board with the data flow paths.

The Counter mode generates events where all the words inside an event are equal. The value of the word is incremented event by event. In the Memory mode the RIMBO injects events to the ROD previously stored in an internal memory through the VME bus. The Counter mode allows burn-in tests since it can inject different events at a very high rate. The maximum trigger rate in this mode is limited by the output data bandwidth. The G-Link protocol is a 16-bit bus clocked at 40 MHz. This protocol provides an output bandwidth of 640 Mbps for each link. Therefore, a typical TileCal event during the commissioning (180 32-bit words) leads to a maximum event rate due to bandwidth limitations of 110 kHz. Table 5.1 shows the bandwidth limitations in terms of maximum trigger rate for different TileCal event types. The different event types depends on the number of samples transmitted for each channels and the number of gains. Two gains per channel are transmitted for calibration events (bi-gain) whereas one gain (auto-gain) is transmitted for physics.

In any of the injection operation modes the CRC is computed for every transmitted event. The result is appended and transmitted as the last word of the event. The CRC can be recomputed offline and checked in order to test the acquisition chain.

Run type	Samples per PMT	Words	Rate(Hz)
Physics	7 Samples, Auto-gain	148	135000
Calibration	7 Samples, Bi-gain	276	72000
Cosmics Commisioning	9 Samples, Auto-gain	180	110000
Calib. Commisioning	9 Samples,Bi-gain	340	59000

Table 5.1: Maximum number of events that can be transmitted by the RIMBO as a function of the event type due to bandwidth limitations.

5.2.4 The ROD Crate

The function of the ROD crate was to hold the RODs under evaluation. The SBC provides VME access to configure and monitor the RODs during the tests. The TBM collects the busy signals from the RODs through the custom P3 VME back-plane and the busy status of the crate is transmitted to the Injection crate to stop the generation of events. Moreover, the TBM receives and distributes the TTC information from the TTC modules in the Injection crate. It provides the signals to synchronize the data and TTC events at DSP level. Each ROD plugged in the crate includes a TM in the rear part of the crate which is used to transmit the processed data to the ROS emulator.

5.2.5 The Readout System Emulator

The ROS emulator consisted of a computer holding two FILAR cards [42] and large storage capacity. The FILAR card is the previous version of the final ROBin [24] cards used in ATLAS to receive the data from the RODs at ROS level. Each FILAR card is able to receive the data from 4 ROLs. Thus, with two FILAR cards we are able to receive the data from 4 RODs. The ROS emulator was able to check the data in real-time or offline. In the former mode the data events coming from the RODs were not stored and only a small fraction of the received events were analyzed. In the offline mode all the received events were stored and analyzed offline. This mode was limited by the storage capacity and could not be used for high rates or very long runs.

5.2.6 ROD Qualification Protocol

Each ROD had to pass all the test levels in order to be validated. The first level, called level 0, was a static test composed of three Diagnostic and Verification System (DVS) tests [43]. These DVS tests basically certified the correct access through VME to every register inside all the programmable devices on the ROD motherboard. Moreover, the correct communication between the Staging FPGA and the OC FPGA was checked by sending several events from the internal memory of the Staging FPGA and reading them out with the OC FPGA. In order to consider the level 0 approved, each ROD had to pass at least three DVS tests. Once a ROD has passed the level 0 tests, 3 levels of dynamic tests level 1, 2 and 3 are applied to the module. In these tests the RIMBO board emulated the FE injecting data to the RODs. The data processed by RODs was stored and checked in the computers. The maximum trigger rate reached by the online check task was approximately 400 Hz. For higher rates, the software could not check all the events online, therefore only a percentage of the processed events were checked. Level 1 test was a single ROD dynamic test at low rate. The trigger rate was 200 Hz and all the events passing across the ROD were checked. After more than 4 hours data processing without errors, the level 1 was approved. At that rate, no busy signals appeared in the ROD system.

Level 2 test was also a single ROD dynamic test, but increasing the injection rate and the number of hours of the run. In that case, the rate of the trigger was 1 kHz and only 40% of processed events were checked. Besides, some busy signals appeared caused by the storage of data coming from RODs. Thus, the correct busy handling was also checked in that test. The ROD had to process data without errors at least during one hour in order to pass the level 2 test.

Finally, the level 3 test was a multiple ROD burn-in test at high rate. In this case, four RODs were tested together during at least 72 hours. The trigger rate was selected to be 1 kHz, and only 10% of the processed events were checked. If no errors were found during the 4 level tests, the ROD was validated and ready to be installed in the ATLAS electronics cavern at CERN.

The TDAQ software adapted for the ROD test-bench can perform two types of data checking to verify the performance of the RIMBO-ROD system. The online checking allows

real time monitoring of the number of errors detected for a reduced number of events whereas the offline analysis performs the error checks to all the data. The ROD were configured in the so-called *copy mode* where raw data is transmitted. In this operation mode the output data should correspond exactly with the data injected by the RIMBO. Error checks are performed in order to qualify the transmission chain:

- The counter operation mode was used for the validation tests. Thus, all the words inside an event should be the same.
- Links coming from different RODs are synchronized. The value of the counter should be equal for all the received events.
- Events should be consecutive for a given link.
- The CRC computed by the RIMBO and appended at the end of each event should match with the CRC computed at ROS level.

A ROD board was validated after passing the complete test protocol that guarantees that every ROD board has been processing data during at least 100 hours. The validation of all the RODs was carried out during four months at the IFIC Valencia lab. A total of 8 x 10^8 events were transmitted by the RIMBO board to the RODs and 5.7 x 10^7 events were checked without errors. The events injected by the RIMBO and processed by the RODs during the production emulated an actual commissioning event in terms of number of bits (180 32-bit words). Thus, we obtain a Bit Error Rate (BER) for the RIMBO-OB-ROD system better than 10^{-10} for a 95% of Confidence Level (CL) (Figure 5.4).

5.2.7 OMB Production and Qualification

The production of the OMB modules was carried out right after the ROD production. The test-bench and the protocol was adapted but the philosophy maintained: the test-bench should emulate the operation conditions as much as possible and every produced board should pass a set of tests in order to become ready for installation. In this case, the TileCal IFIC-Valencia group led the fabrication and assembly of the modules although they were carried out in external companies. The first batch of boards were delivered in autumn of 2008 although a

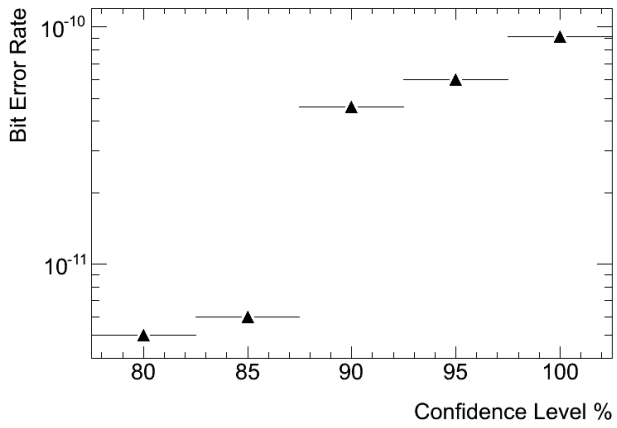


Figure 5.4: BER as a function of the Confidence Level. Note that a BER better than 10^{-10} is obtained for a 95% of C.L.

fully operative prototype was produced one year before allowing the developing and test of the firmware. The boards were programmed and the VME access checked using the static tests implemented in the dedicated IGUI. The static tests essentially certified the correct communication between the SBC and the OMB internal registers. Once the board has passed the static tests, a set of dynamic tests were performed. In this case, the test-bench used to validate the modules consisted of a trigger crate, a processing crate and the ROS emulator (Figure 5.5). Two OMB boards in injection mode were used to emulate the signals coming from the detector. A third OMB in checking mode was receiving these signals and transmitting the events after the CRC check to a ROD module which was configured to transmit the raw data to the ROS system. Therefore, the CRC appended to each event by the two OMBs in injection mode was checked at the end of the processing chain to verify the correct data flow. Moreover, in order to guarantee that the OMB in checking mode was selecting the correct link in case of CRC errors, the error generation mode was enabled. In this mode, already explained in Chapter 4, for every L1A signal received the OMB injects events with CRC errors in half of the output links . With two OMBs in injection mode and the correct routing of fibers we can have for each couple of links (emulating the two links of a Interface Board) one link with CRC error and the other one free of errors.

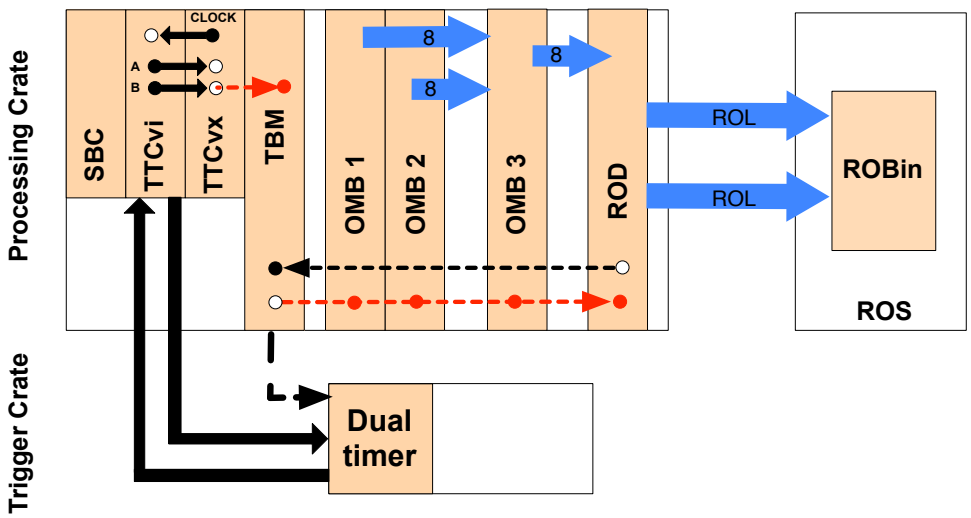


Figure 5.5: Sketch of the OMB production test-bench at IFIC-Valencia. Solid black arrows represent the generation of L1A signals. Dashed black arrows shows the propagation of busy signals. Red arrows show the transmission of the TTC information and blue arrows the flow of data events. In the picture, OMB 1 and 2 are configured in the injection mode whereas OMB 3 is in checking mode.

5.3 Installation and Commissioning of the ROD System

The installation of the ROD modules in the ATLAS electronics cavern (USA15) started during the summer of 2005 simultaneously with the installation of the detector modules (Figure 5.6). First, the VME crates and services were installed in the racks. The ROD modules were installed in four 9U VME crates corresponding to the four TTC partitions. Therefore, four associated 6U VME crates were installed close to each crate to manage the TTC signals of the partition. The eight crates were installed in two racks. A third rack in the middle houses the TTCoc was used to distribute the TTC signals to the front-end modules and the patch panel to connect the fibers coming from the front-end to the ROD input fibers. The racks provide vertical air cooling to the VME crates with a turbine in the top of the rack and water heat exchangers between the crates (Figure 5.7). In order to guarantee a proper air flow

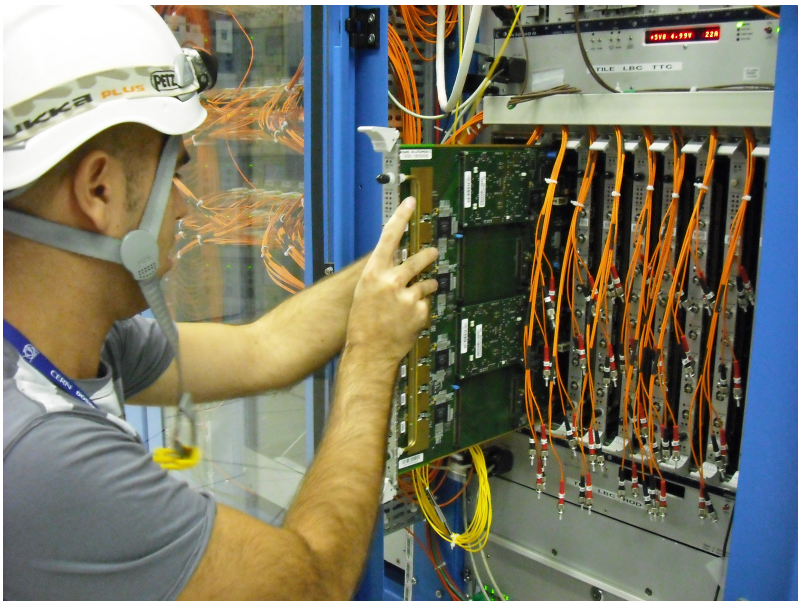


Figure 5.6: Installation of the ROD modules in USA15.

through the VME crates it was decided to use separated power supplies. In particular, air cooled power supplies placed in the rear part of the crate are used for the TileCal ROD crates (Figure 5.8).

First, the laboratory setup was replicated in USA15 which essentially consisted of a RIMBO module to emulate the signals from the detector plus an OB to replicate the signals of the RIMBO. The data generated was injected into the first installed RODs and the processed data stored in a ROS prototype. As soon as the front-end modules were installed and available for commissioning, the fibers coming from the RIMBO were replaced by fibers coming from the actual detector. The OB were used in this second stage to replicate the signals from few front-end modules in order to feed several ROD boards. In this configuration, the RODs were used to qualify and validate the installation of the front-end modules. The installation of RODs was carried out in parallel with the installation of front-end modules. The installation of the ROD system including the TTC modules, cabling and fiber was completed in spring of 2007 (Figure 5.9).

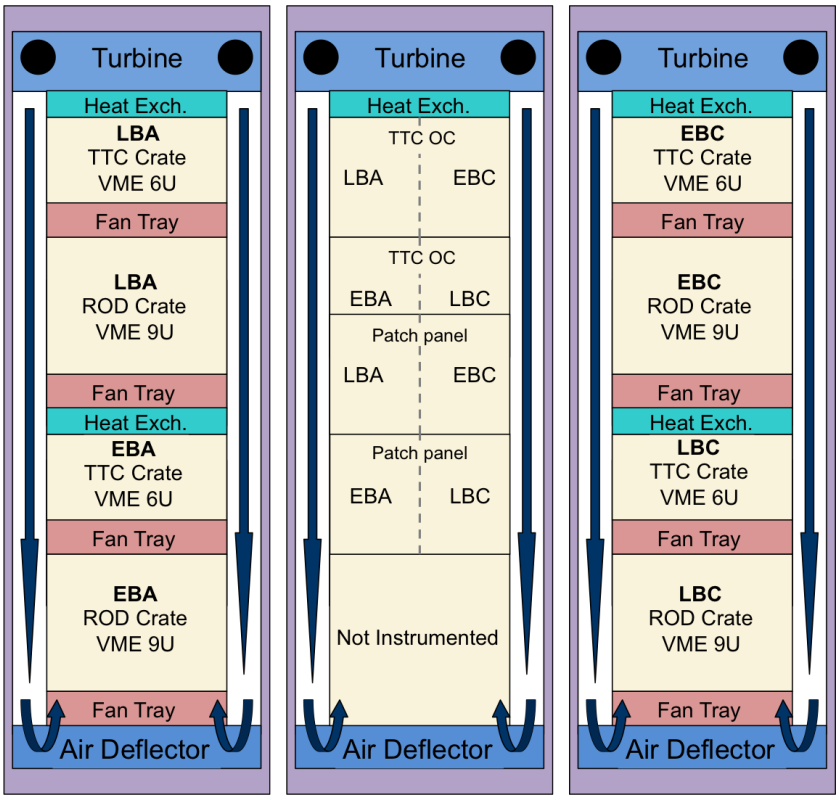


Figure 5.7: Drawing of the USA15 racks housing the ROD (left and right racks) and TTCoc (middle rack) crates.

5.3.1 TileCal Commissioning

Before starting the installation of the TileCal front-end modules in the ATLAS cavern, the barrel and one extended barrel cylinders were fully assembled on the surface. The goal was to certify the assembly before lowering the modules in the ATLAS pit and it consisted in several tests and certification of extraction tools and support, deformation measurements, simulation of the LAr electromagnetic calorimeter, weight with load tests, etc.

The first fully instrumented modules (i.e., with optics and all the certified front-end electronics inserted inside) were installed in the ATLAS cavern in February 2004. The installed modules and superdrawers electronics were qualified and validated during the installation of the detector. The work was distributed in five teams working in parallel to verify the proper

Figure 5.8: Picture of the rear part of a ROD 9U VME crate with the water cooled power supply and the TM.

functioning of the complete system. Team 1 was responsible of the installation of the detector parts and services. Team 2 carried out the reparation and maintenance of the installed parts. Team 3 were responsible for the cabling and connectivity between USA15 and the detector. Team 4 was acquiring calibration data for the installed modules to verify the behavior of the electronics. Finally, Team 5 analyzed the data acquired in order to detect problems and provide feedback to Team 2. The installation of the detector modules finished in 2006. The central barrel was assembled in one side of the ATLAS cavern and then slid to its z=0 position in the center of the detector. Figure 5.10 shows a extended barrel assembled and being slid in the ATLAS cavern.

Nevertheless, after the LHC incident in September 2008 (see Section 5.4.1) it was decided to proceed with a maintenance campaign in the front-end electronics. In the so-called refurbishment period some superdrawers were extracted and the detected weakest parts of the system were repaired or reinforced. Around 32% of the front-end electronics superdrawers were opened and repaired, and 4.2% of Low Voltage Power Supplies (LVPS) were replaced

Figure 5.9: Picture of the TileCal back-end system racks in USA15.

or repaired. After this maintenance program, 98.74% of the channels, or 99.07% of the cells, were operational. Moreover, that operation improved the reliability of the system during the first years of operation.

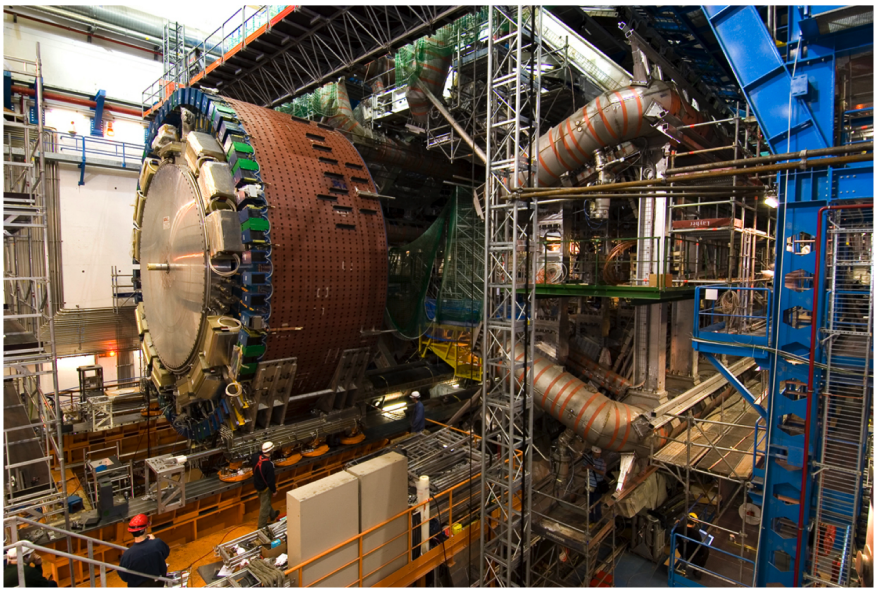


Figure 5.10: Picture of the a TileCal extended barrel assembled in the ATLAS cavern after installation.

5.3.2 Integration of the ROD System to the Combined Cosmic Ray Muons Data Taking

As soon as the different sub-detectors were arriving to the final stage of their installation and commissioning, the first tests for the integration of the ATLAS sub-systems started. These tests were carried out in the so-called Milestone Weeks. The Milestone Weeks were dedicated to integrate the sub-systems in order to operate the experiment as a whole, from sub-detector to permanent data storage, with an increasing number of sub-detector systems involved at each stage. The main goal was to exercise with the available part of each sub-system and try keep it in stable running conditions for many hours to emulate the final operation conditions.

The first Milestone Week was held early 2007 and only the barrel calorimeters (TileCal and LAr) participated. The first cosmics rays acquired by the TileCal RODs were processed during this week. A special trigger provided by TileCal was used. The following Milestone Weeks were including the other sub-systems in subsequent steps. Therefore, TileCal and part of the LAr calorimeter were the only sub-systems that participated in all the Milestone Weeks.

The last one was held in the summer of 2008 where all the ATLAS sub-detectors participated in the acquisition of cosmic rays proving the readiness of ATLAS for the coming first LHC collisions in September 2008 (Figure 5.11).

The rate of cosmic rays hardly reached few Hz. This very low rate of events allowed the RODs to transmit both the reconstructed data (used in the HLT) and the complete raw data from the detector (Fragment 0). The raw data was used for offline reconstruction and used as reference to validate the online reconstruction. Moreover, the cosmic rays were not synchronized with the global ATLAS clock and the phase of the PMT pulses produced by the particles was not fixed. Therefore, the iterative Optimal Filtering (described in next Chapter) was used to reconstruct these signals in the ROD DSP.

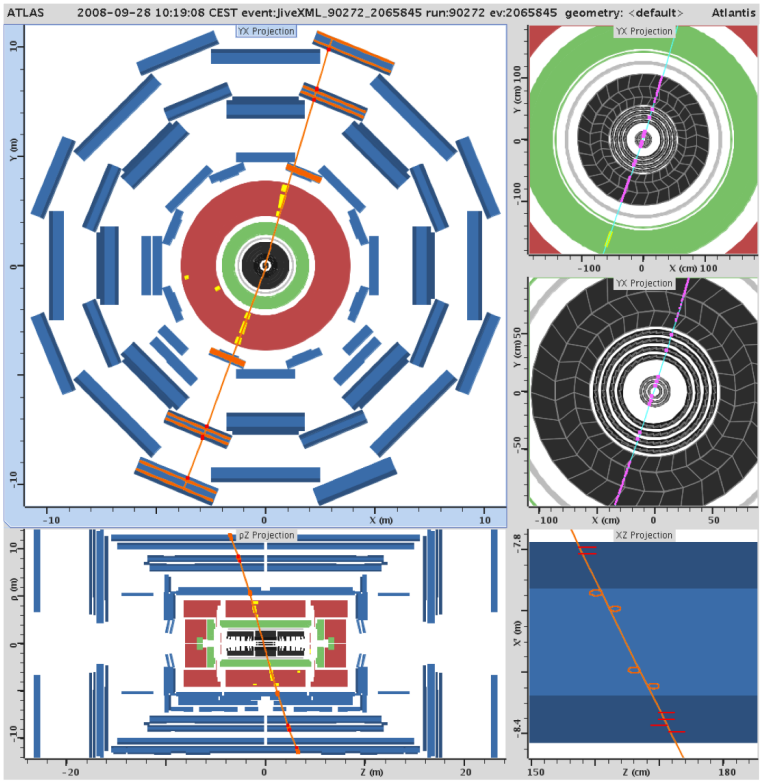


Figure 5.11: Snapshot of the ATLAS Event Display with a nice cosmic crossing the detector acquired in an ATLAS combined run in September 2008.

In order to guarantee the readiness of the ATLAS acquisition system for the first years of LHC operation some Level 1 Trigger high rate tests were performed. The triggered cosmic rays were mixed with some random triggers to increase the Level 1 Trigger rate to some tens of kHz. The TileCal ROD system performed stably with absence or negligible percentage of dead-time up to 45 kHz transmitting reconstruction plus complete raw data and close to the nominal 100 kHz rate transmitting only the reconstruction fragment. The compressed raw data fragment was implemented during the first years of operation to handle rates above 45 kHz by transmitting only a portion of the PMT samples. The cosmic ray data acquired was used to validate the performance of the online reconstruction in comparison with the offline reconstruction. Moreover, it was also used to synchronize the readout in time using the cosmic muons. The method consisted on a comparison between the expected time-of-flight of muons crossing the detector and the reconstructed time. This was the first estimation of the timing constants needed to compute the Optimal Filtering weights, which were optimized later, first using single beams and finally with collisions data.

5.4 Performance in ATLAS Operation

5.4.1 Readiness of ROD System for Collisions in 2008

The first beam was injected in the LHC on September 10th of 2008. The beams circulated in both directions and they were dumped in the ATLAS collimators in the so-called splash events. The ROD system performed perfectly collecting the few splash events provided by LHC. These events were used to improve the timing constants previously estimated using cosmic muons and laser calibration data. In single beam data, the signal timing was investigated as a function of z (beam direction), and the time-of-flight was calculated in the lateral direction. The laser corrections which take into account the laser and WLS fiber lengths were applied for the timing calculation. In addition, since the timing constants were meant to be used for particles produced in collisions in the Interaction Point (IP), one more correction was applied which considers the fact that beam halo particles were not coming from the IP but traveling along the beam direction. This procedure was used in the subsequent LHC single beam splashed events at the beginning of LHC collisions periods. Figure 5.12 shows the absolute

cell time as a function of z (beam direction) for a beam 1 splash event. The time of the particles crossing a cell increase with the distance between the cell and the collimator. After applying the time-of-flight correction that assumed a track parallel to the z-axis we obtain a flat distribution demonstrating the very good time equalization.

However, the LHC machine was fully commissioned to operate at 5 TeV in 7 out of 8 sectors and the remaining sector was commissioned up to 4 TeV. The commissioning of this last sector at 5 TeV started few days after a suddenly magnetic field quenching was detected that resulted in a helium leakage into the insulation vacuum of the cryostat which was finally released to the LHC tunnel. The incident occurred on September 19th of 2008 and the out of control increase of the pressure lead to damages in several LHC sub-sectors. It implied one more year for detector commissioning which in the case of TileCal was used to repair and improve the reliability of the front-end electronics. The ROD system was ready for collisions since 2008 and only few firmware updates were introduced during the following years. The ROD system was used to collect calibration data to certificate the operations in the front-end electronics.

5.4.2 Operation During Early LHC Collisions (2009-2010)

LHC restarted the beam operation on November 20th of 2009. Again, some single beam splash events were provided by the LHC machine and were used to adjust the timing corrections computed one year before. Three days later, the first proton-proton collisions at the injection

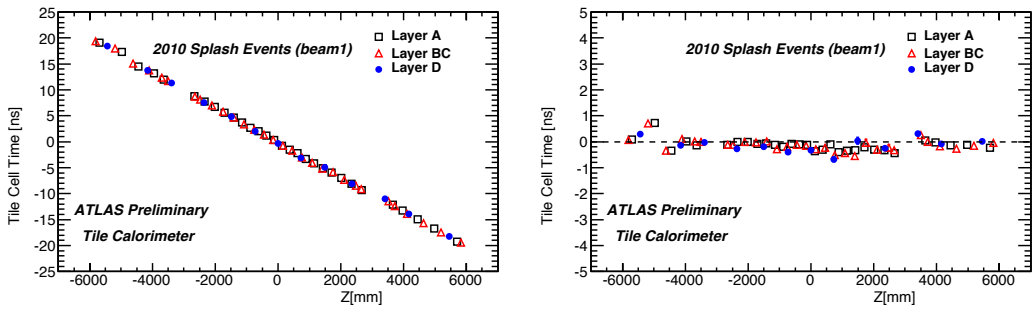


Figure 5.12: Time of TileCal signals recorded with single beam data on February 2010 before (left) and after (right) time-of-flight corrections.

energy of 450 GeV per beam were recorded by the ATLAS detector and only few days later with an energy of 1.18 TeV per beam. The LHC was fully commissioned at the end of the year although non significant stable beams were provided for collisions in the detectors. In 2010 the LHC started to produce luminosity for the experiments at a 7 TeV center of mass energy. The instantaneous luminosity during this year was continuously increased from an initial value of 0.0016×10^{30} cm^{-2}s^{-1} (with 1 colliding bunch) until a final maximum peak luminosity of 2.1×10^{32} cm^{-2}s^{-1} (with 348 colliding bunches in the ring and a bunch spacing of 150 ns). The peak mean value of the number of interactions per bunch crossing, $< \mu >$, evolved from $< \mu >$=0.0142 to $< \mu >$=3.31 along the year.

The low instantaneous luminosity of LHC during the first part of 2010 produced a very low Level 1 trigger rate even if extremely low thresholds for the trigger were used. The TileCal ROD system performed correctly transmitting complete raw and reconstruction data. Moreover, the extremely low Level 1 rates and the absence of pile-up allowed the usage of the Optimal Filtering iterative algorithm as the default reconstruction method in the DSP. The maximum Level 1 trigger rate of 40 kHz allowed by this ROD operation mode was reached due to the increase of the luminosity, and the TileCal deadtime, which was increased substantially. In addition, the complex and simple deadtime parameters of the CTP were tuned to handle with the higher luminosities producing very high instantaneous trigger rates. Some DSP firmware updates were needed to cope with this new trigger structure and the deadtime from the TileCal RODs was maintained in the levels of other sub-detectors (Figure 5.13) representing a 10% of the total ATLAS dead-time. Another source of inefficiency in the data acquisition was the stop and restart of a run during stable beams. The stop time due to TileCal problems represented the 3.6% of the total stop time in ATLAS during 2010.

A total of 48.1 pb^{-1} were delivered by the LHC in 2010 with 45.0 pb^{-1} recorded by ATLAS. This represented a total data taking efficiency of 93.5%. The 2010 data taking can be considered as part of the commissioning period for the detectors because the total integrated luminosity delivered by the LHC during the entire year (48.1 pb^{-1}) represented only few hours of data taking during the following years.

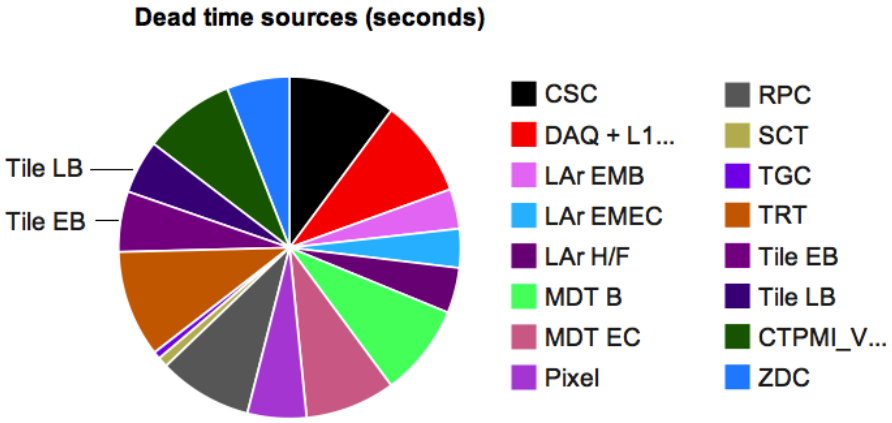


Figure 5.13: Percentage of ATLAS deadtime per sub-detector during 2010. The TileCal contribution represented the 10% of the total ATLAS dead-time.

5.4.3 Operation and Performance at Design Specifications (2011-2012)

The performance of the ROD system during 2010 was satisfactory although became evident that it was working very close to the limit in terms of Level 1 trigger rates. During the last period of the year the output bandwidth of the ROD was eventually saturated producing undesired deadtime. These results and the expected increase in the LHC peak luminosity requited a change in the ROD configuration for 2011. Essentially, the Optimal Filtering non iterative method was selected as the default reconstruction algorithm in the DSP and the complete raw data fragment was replaced by a first version of the compressed raw data fragment. The non iterative method reduced the processing time with respect to the iterative whereas the compressed raw data fragment reduced the size of the output data fragment of the ROD to avoid bandwidth saturation. The threshold of the compressed raw data fragment was initially set to 5 ADC-counts meaning that only those samples with pulses above threshold were transmitted out of the ROD. The 2011 LHC startup parameters were very similar to 2010 but they were increased to values close to the nominal ones at the end of the year. The energy was maintained at 3.5 TeV per beam. The maximum number of colliding bunches was increased up to 1331 with a structure of 12 trains and with a minimum bunch spacing

of 50 ns. The maximum peak luminosity was increased up to 3.6516 x 10^{33} $cm^{-2}s^{-1}$ with a peak $< \mu >$=15. The reduction of the bunch spacing up to 50 ns and the increase of $< \mu >$ produced an increase of the pileup and the default method for offline reconstruction was also changed to the Optimal Filtering non-iterative.

During 2011 the total integrated luminosity delivered by the LHC in ATLAS was 5.61 fb^{-1} with 5.25 fb^{-1} recorded by ATLAS. The data acquisition efficiency was 93.6% where again the main sources of the inefficiency were the deadtime from the different systems, the stop and restart of runs and the warm start which represents the time between the declaration of stable beams by the LHC and the actual start of acquisition of data. The ROD system contributed considerably to the deadtime representing the 13.61% of the total deadtime in ATLAS during 2011 (Table 5.2). On the other hand, the TileCal ROD system never caused a stop of a run with stable beams during 2011. The deadtime introduced by the ROD system was mainly caused by two reasons. First, the saturation of the output bandwidth of the ROD due to an increase in the number of channels above the compressed raw data threshold. The threshold was increased from 5 to 6 ADC-counts to reduce the number of samples in the compressed raw data fragment and the deadtime from TileCal was considerably reduced. However, the other source of deadtime from the TileCal RODs was caused by the front-end LVPS. It was found that SEU in one device in the LVPS produced an automatic switch-off of the LVPS of a front-end superdrawer. Around one LVPS was switched-off in TileCal by a SEU every 1 pb^{-1} and therefore the number of trips increased with the luminosity. A software mechanism to recover the LVPS was implement which needed around 2 minutes to complete reestablish the power in the front-end superdrawer. Fortunately, in terms of data quality this failure is recoverable since represented a small area of the detector. However, it affected the acquisition efficiency of the ROD system. The DSP synchronization task assumes that a data event is received from every superdrawer for each L1A signal. If a superdrawer stops sending data during a run, a timeout mechanism is executed in the DSP and after some time the DSP generates a fake event. This mechanism consumes some extra processing time and was the main cause of TileCal deadtime during 2011.

The timeout mechanism was improved in mid 2012 to reduce the deadtime produced by the increasing number of trips. The recovery software was updated to change the DSP

System	Percent
RPC	8.27
CTPMIVME	8.16
CSC	7.66
TRT	7.65
TileEB	**7.39**
SIMPLE	7.28
COMPLEX	7.22
MDT EC	7.20
LAr EMEC	7.18
MDT B	6.99
TileLB	**6.22**
LAr EMB	5.53
Pixel	4.67
DAQ + L1Calo	4.33
LAr H/F	4.21

Table 5.2: Percentage of the integrated deadtime per system for 2011. TileCal represents the 13.61% of the total deadtime. Simple and complex deadtime corresponds to a trigger mechanism to limit the Level 1 rate and to prevent two very consecutive L1A trigger.

configuration during the recovery of a LVPS that tripped. The DSP automatically introduced empty events for the corresponding superdrawer which accelerated the processing of events. Moreover, it avoided the processing of corrupted data produced by modules being recovered.

Data Taking During 2012

The LHC conditions continued developing during 2012. First, the energy was increased up to 4 TeV per beam. The instantaneous luminosity was also increased with respect to the previous years with a maximum value in 2012 of 7.31×10^{33} cm^{-2}s^{-1} (Figure 5.14), very close to the LHC design value of 10^{34} cm^{-2}s^{-1}.

The number of bunches was slightly increased up to 1368 whereas the bunch spacing was kept at 50 ns (Figure 5.15). The decision of keeping 50 ns of bunch spacing but increasing the peak luminosity lead to an increase also in the number of interactions per bunch crossing. The maximum mean number of events per crossing was above 30 during most time in 2012.

These machine settings allowed to deliver a total integrated luminosity of 23.3 fb^{-1} during 2012 with 21.7 fb^{-1} recorded by ATLAS representing a data taking efficiency of 93.1%. In the

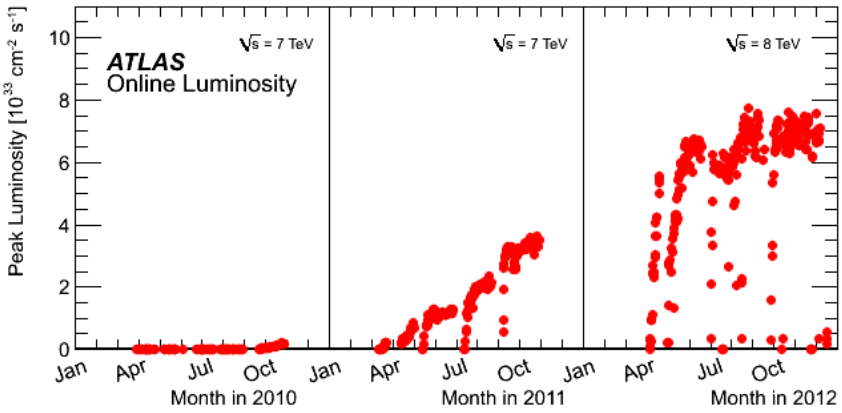


Figure 5.14: The peak instantaneous luminosity delivered to ATLAS per day versus time during the p-p runs of 2010, 2011 and 2012.

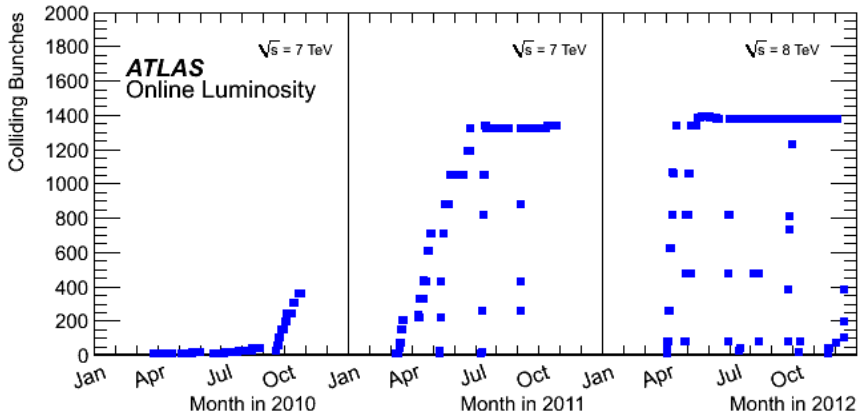


Figure 5.15: The number of colliding bunches in ATLAS versus time during the p-p runs of 2010, 2011 and 2012.

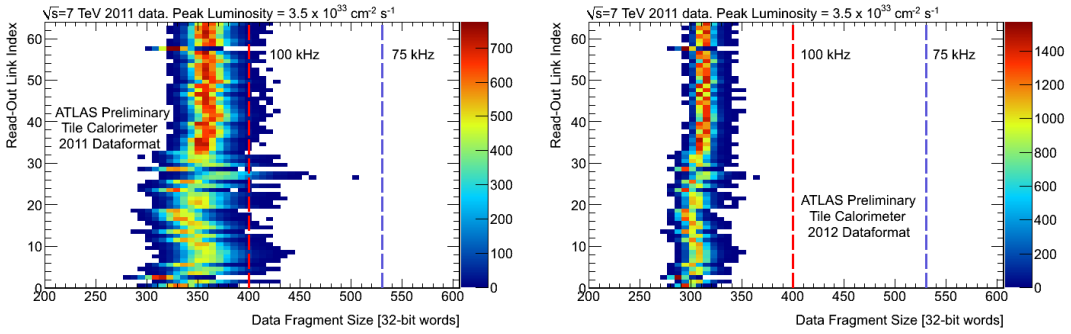


Figure 5.16: ROD output fragment size in 32-bit words for each Readout Link for the data format used in 2011 (left) and the compressed format deployed in 2012 (right).

TileCal ROD system an optimized version of the compressed raw data fragment (discussed in Chapter 3) was deployed (Figure 5.16). It allowed to decrease the selection threshold to 5 ADC-counts which was maintained during the entire year without any observed bandwidth saturation even at the highest luminosities and pileup. The new compressed data format also permitted to run at the nominal Level 1 trigger rate of 100 kHz without experiencing any bandwidth saturation in the output of the ROD.

The contribution of the TileCal back-end system to the total ATLAS data acquisition deadtime during 2012 was 9.9% (5% for LB and 4.9% for EB, see Table 5.3). However, it was important to study this contribution for the first and second half of the year since some critical developments were installed in the summer technical stop in order to reduce the deadtime from TileCal. The timeout logic was improved and the size of the DSP input buffers increased in order to handle the new Level 1 Trigger simple and complex deadtime settings. The result is that the deadtime from TileCal represented the 12.0% of the total ATLAS deadtime before the summer technical stop whereas it was reduced to the 8.6% after the implementation of the new DSP code. Moreover, a large part of the deadtime in the second half of the year was produced by PLL unlock problems in the Laser system.

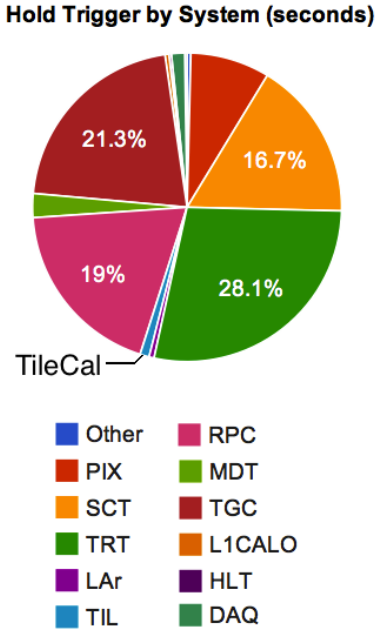
Furthermore, a new problem in the acquisition system turned up during spring. For about every 1 fb^{-1} acquired, data stopped flowing through a group of ROLs in one of the EB partitions. A clear reason for the problem was not found although it was correlated to the

System	Percent
COMPLEX	16.2
RPC	15.4
CTPMIVME	5.9
TRT	5.7
MDT B	5.6
MDT EC	5.4
SIMPLE	5.3
CSC	5.2
SCT	5.2
Tile EB	**5**
DAQ + L1Calo	4.9
Pixel	4.9
Tile LB	**4.5**
LAr EMEC	4.5
TGC	3.1
LAr H/F	1.9
LAr EMB	1.4
BCM	0.1
CTPLUCID	0.1
MUCTPI	0
ZDC	0
DAQ	0

Table 5.3: Percentage of the integrated deadtime per system for 2012. TileCal represents the 12% of the total deadtime.

automatic reconfiguration of the front-end electronics performed after a LVPS trip. The large traffic of commands through the TTC network might be the cause of the problem, which only appeared 7 times before the summer technical stop. However, the acquisition got stuck and the intervention of an expert was needed, with the consequence of a tenths of minutes of data taking lost. Sometimes, the stop and restart of the run was also needed to restore the acquisition, which increased even more the deadtime. During 2011 TileCal never forced the stop of a run with declared stable beams but it represented the main cause in the first part of 2012.

During the summer technical stop an automatic recovery procedure to detect a ROL problem, reset the components in the data path and restart the acquisition without human intervention was implemented in the TileCal DAQ software. Essentially, the recovery time



System	Percent
TRT	28.1
TGC	21.3
RPC	19
SCT	16.7
PIX	8.3
MDT	2.5
DAQ	1.4
TILE	**1**
LAr	0.5
L1CALO	0.4
Other	0.4
HLT	0.3
L1CT	0.1
BCM	0.1
CSC	0
LUCID	0
ZDC	0
ALFA	0

Figure 5.17: Hold trigger time per sub-system in 2012. TileCal represents the 1% of the total time.

was reduced from tenths of minutes to few seconds. The problem continued appearing the rest of the year although the automatic procedure always worked properly. After the technical stop, TileCal was the reason to stop a run with declared stable beams only 2% of the total time and again it was caused by PLL unlock in the laser ROD.

Due to the large deadtime introduced by the stop of runs during stable beams, a new TDAQ mechanism was introduced to restart only one specific sub-system. During the so-called *TTC restart* command, the trigger is also in hold by the sub-system being recovered introducing a deadtime in the acquisition. The trigger is also in hold during the automatic stopless ROL recovery procedure. The trigger hold time introduced by TileCal either because of a *TTC restart* or during the stopless ROL recoveries represented only the 1% of the total hold trigger time in ATLAS (Figure 5.17).

In general, the software and firmware updates introduced in the back-end system during the summer technical stop improved the TileCal data acquisition efficiency considerably in the

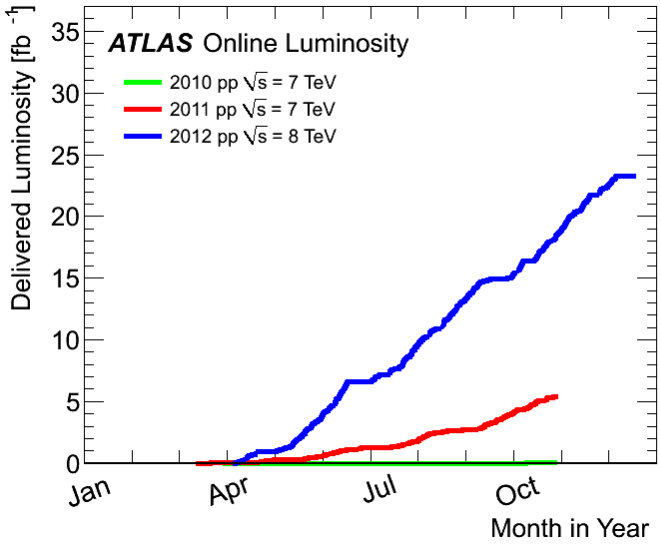


Figure 5.18: Cumulative luminosity versus day delivered to ATLAS during stable beams and for p-p collisions. This is shown for 2010 (green), 2011 (red) and 2012 (blue) running.

second part of the year where a large fraction of the total luminosity was delivered. Figure 5.18 shows the evolution of the integrated luminosity during 2010, 2011 and 2012. The delivery luminosity during 2010 is negligible in comparison with 2011 and 2012. In the case of 2012 the plateau in June corresponds to the technical stop and therefore most of the luminosity of 2012 was delivered after the technical stop with the DSP and software improvements already installed.

Chapter 6

The ROD Reconstruction Algorithms

6.1 Introduction

The digitized samples generated by the ATLAS hadronic tile calorimeter contain information about the energy deposited by a particle as it travels through the calorimeter. In order to extract this information from the samples, three different algorithms were studied, the Flat Filtering algorithm, the Fit method and the Optimal Filtering algorithm. These three algorithms calculate the deposited energy, the former by estimating the signal area, and the latter two by estimating the signal amplitude.

However, the best energy resolution is achieved with the Fit and Optimal Filtering methods although the Optimal Filtering is much simpler to implement and faster. Thus, Optimal Filtering is the default algorithm used to reconstruct the energy in real time running on the Tile Calorimeter ROD boards, where the processing time is one of the big constraints.

The Optimal Filtering algorithm is designed to account for the LHC conditions where the digitization clock is synchronized with the trigger clock, thus the signal samples have always a fixed phase with small variations [44]. Moreover, the algorithm discriminates the energy depositions with large phase differences with respect to the expected one. This feature is

important to decrease the effect of the out of time pileup. During cosmic muon data taking, the phase of the energy deposition is not fixed. In this case an iterative Optimal Filtering method can be used. The first two iterations are used to estimate the phase of the pulse which is used to select the correct set of weights used in the third and final iteration where the energy is calculated. This operating mode was used during the commissioning period to reconstruct the signals of cosmic muons and during the first year of LHC operation without out of time pileup.

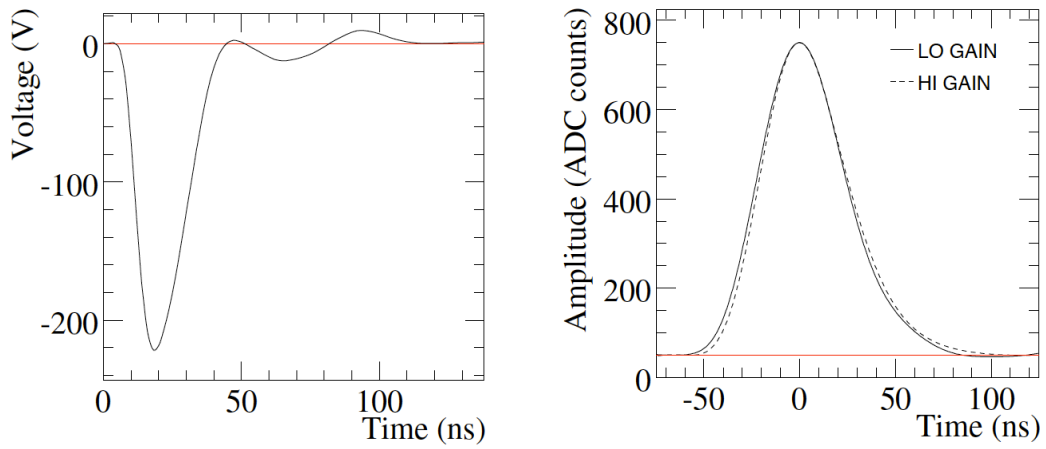
In addition to the energy reconstruction at PMT level, each of the TileCal ROD modules have to provide a special fragment for the total energy and its transverse and Z projections. This information is used as a E_T^{miss} trigger in the HLT system.

This chapter describes the Optimal Filtering algorithm, computation of the weights, calibration of the energy and iterative methods used for special running conditions. Then, the implementation of the signal reconstruction algorithms in the DSPs are detailed, highlighting the principal constraints and limitations of the online reconstruction.

6.2 Optimal Filtering Algorithm

The TileCal PMT analog pulses (Figure 6.1) are shaped and digitized in the front-end electronics using the 40 MHz clock synchronized with LHC bunch crossings. For each event selected by the first level of trigger, 7 samples, S_i, are transmitted to the back-end system where the energy depositions are reconstructed in the RODs. The digital filters used to reconstruct the signals take advantage of the knowledge of the pulse shape from the electronics to reduce the noise contribution and to determine the time of the deposition. Although the pulse shape is slightly depending on the type of particle crossings the scintillator and the amount of energy deposited, the average pulse shape is used to reconstruct and simulate the depositions (Figure 6.1).

The pulse shapes normalized to unit amplitude, $g(t)$, were obtained separately for high gain (HG) and low gain (LG), and for physics and calibration data. Figure 6.1 shows the normalized pulse shape for physics events for HG and LG signals. The method used to obtain the normalized pulse shapes used real energy depositions with different amplitudes



(a) (b)

Figure 6.1: (a) Analog pulse produced by a PMT for a Laser pulse. (b) Physics signal pulse shapes used as reference for the Optimal Filtering weights calculation. The pulse shapes are slightly different for high and low-gain. They were obtained from test beam data and are available in the Athena software.

and crossing the detector at different times. Then, each signal is normalized and shifted to peak at a fixed time. Finally, overlaying the signals from different depositions provides the complete reconstruction of the pulse shape.

The pedestal is defined as the baseline of the signal (Figure 6.2). Typical values are between 30 and 60 ADC-counts, and is adjustable from the TileCal front-end electronics. Figure 6.3 shows the average value and RMS of the pedestal for the HG and LG of all the channels in a typical TileCal module.

The phase reconstructed with the Optimal Filtering is the time difference between the peak and the expected arrival time of the pulse. This expected time is estimated and stored in the conditions database for each channel. It is measured with respect to the central sample and is used to compute the weights of the Optimal Filtering algorithm (see Section 6.2.1).

The amplitude of the pulse is the other parameter of interest as it is proportional to the energy deposited in a calorimeter cell. It is defined as the pulse height subtracting the pedestal. In TileCal, two sources of noise distort the signal:

- Thermal noise. Thermal noise is mainly intrinsic to the electronics, it consists of small

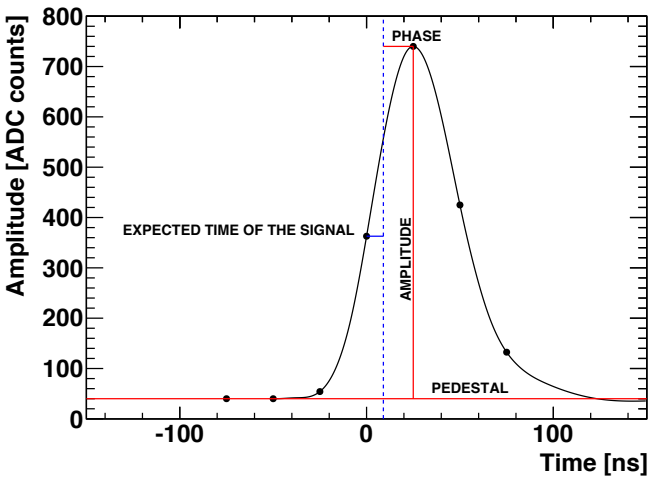


Figure 6.2: Physics pulse shape with the definition of amplitude, reconstructed phase and pedestal. The points represent the seven samples transmitted to the ROD.

variations of the samples around their expected value. The electronic noise was characterized first during the testbeam and later during the commissioning period. Non-Gaussian tails were found in the energy distributions of the channels for pedestal runs. The noise was finally described using a double Gaussian distribution showing a good agreement between the real and MC simulated noise [45]. The variation of the pedestal are mainly produced by thermal noise and the contribution of the electronic noise is larger for HG than for LG (Figure 6.3). The characterization of the thermal noise throughout the detector showed a sigma of the distribution around 1.4 ADC-counts for the HG whereas 0.6 ADC-counts for the LG.

- Pile-up events. Pile-up noise is due to uncorrelated events that are produced close in time to the event of interest and consequently they are piled-up with it. Thus, pile-up noise depends strongly on the LHC conditions, in particular instantaneous luminosity which strongly influences the number of collisions.

The reconstruction algorithms used in TileCal should properly reconstruct the parameters of the signal (amplitude and phase). The contribution of the noise should be minimized in

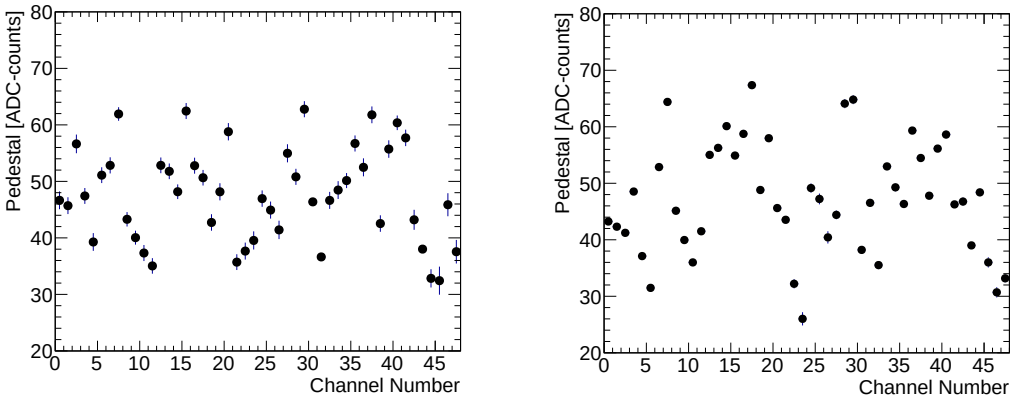


Figure 6.3: Mean value and RMS of the pedestal as a function of channel ID for LBA01 in pedestal run 216276 for HG (left) and LG (right).

order to eliminate any bias on the reconstruction. The Optimal Filtering algorithm calculates the amplitude and phase by means of a weighted sum of the digitized samples, using weights that minimize the contribution of the noise. There exist two flavors of Optimal Filtering, OF1 and OF2 [46]. The difference between them is that, the so-called OF1, reconstructs the amplitude and phase of the samples and estimates the pedestal from the first sample. On the other hand, OF2 calculates the amplitude, phase and pedestal as parameters of the reconstruction. The two methods are implemented in the TileCal RODs but only OF2 is used for ATLAS operation.

The pulse produced by the TileCal front-end electronics can be expressed with a functional form as:

$$S(t) = Ag(t - \tau) + p \quad , \tag{6.1}$$

where $g(t)$ represents the pulse shape normalized in amplitude as a function of time t. A is the amplitude of the signal, τ the relative phase and p the pedestal level (Figure 6.2). The pulse is digitized every 25 ns to obtain n samples,

$$S(t_i) = Ag(t_i - \tau) + p \quad \text{for} \quad i = 1, ..., n \ . \tag{6.2}$$

The phase of the pulse can be defined as the time difference between the central sample and the peak of the pulse. However, the phase obtained by Optimal Filtering depends on the definition of the weights used. If the weights are calculated for $g(t_{\text{central}} = \tau_0 = 0 \text{ ns}) = 1$ then the phase provided by the Optimal Filtering corresponds to our definition of phase τ. But, if the weigths have been calculated for any other phase, *i.e.* $g(t_{\text{central}} = \tau_0 \neq 0 \text{ ns}) < 1$, then the phase provided by the Optimal Filtering is $\tau - \tau_0$. Therefore when the phase of the samples is $\tau = \tau_0$ (same phase as for the weights), then the Optimal Filtering should return a phase equal to 0 ns. Essentially, the phase reconstructed by Optimal Filtering represents the time between the expected phase of a pulse produced by a particle coming from the interaction point (which is used to compute the weights) and the actual peak of the reconstructed pulse.

At the LHC, phase variations are expected to be very small, therefore a Taylor expansion can be applied in order to linearize the dependence with τ:

$$S(t_i) \approx Ag(t_i) - A\tau g'(t_i) + p \quad \text{for} \quad i = 1, ..., n \ , \tag{6.3}$$

For simplicity, the element at t_i can be re-written with the subindex i:

$$S_i \approx Ag_i - A\tau g'_i + p \quad \text{for} \quad i = 1, ..., n \ , \tag{6.4}$$

where S_i represents the digitized sample at t_i. An additional term n_i should also be included to take into account the noise contribution:

$$S_i \approx Ag_i - A\tau g'_i + p + n_i \quad \text{for} \quad i = 1, ..., n \ , \tag{6.5}$$

where by hypothesis $\langle n_i \rangle = 0$.

We can now define three quantities, u, v and w, such as:

$$u = \sum_{i=1}^{n} a_i S_i \ , \qquad v = \sum_{i=1}^{n} b_i S_i \quad \text{and} \qquad w = \sum_{i=1}^{n} c_i S_i \ , \tag{6.6}$$

and we require the expected value of these quantities to be equal to A, $A\tau$ and p, respectively:

$$\begin{aligned} A &= \langle u \rangle = \langle \textstyle\sum_{i=1}^{n} a_i S_i \rangle = \textstyle\sum_{i=1}^{n} a_i \langle S_i \rangle \quad , \\ A\tau &= \langle v \rangle = \langle \textstyle\sum_{i=1}^{n} b_i S_i \rangle = \textstyle\sum_{i=1}^{n} b_i \langle S_i \rangle \quad , \\ p &= \langle w \rangle = \langle \textstyle\sum_{i=1}^{n} c_i S_i \rangle = \textstyle\sum_{i=1}^{n} c_i \langle S_i \rangle \quad . \end{aligned} \tag{6.7}$$

Combining Equation (6.5) with Equation (6.7) and imposing minimum variance of u, v and w with respect to A, $A\tau$ and p, respectively, we obtain a set of equations to calculate the optimal weights a_i, b_i and c_i. Hence the $n+3$ equations for the amplitude weights are:

$$\begin{aligned} &\textstyle\sum_{i=1}^{n} a_i g_i = 1 \quad , \\ &\textstyle\sum_{i=1}^{n} a_i g_i' = 0 \quad , \\ &\textstyle\sum_{i=1}^{n} a_i = 0 \quad , \\ &\textstyle\sum_{j=1}^{n} a_j R_{ij} - \lambda g_i - \kappa g_i' - \nu = 0 \quad \text{for} \quad i = 1, ..., n \quad . \end{aligned} \tag{6.8}$$

The $n+3$ equations for the phase weights are:

$$\begin{aligned} &\textstyle\sum_{i=1}^{n} b_i g_i = 0 \quad , \\ &\textstyle\sum_{i=1}^{n} b_i g_i' = -1 \quad , \\ &\textstyle\sum_{i=1}^{n} b_i = 0 \quad , \\ &\textstyle\sum_{j=1}^{n} b_j R_{ij} - \mu g_i - \rho g_i' - \phi = 0 \quad \text{for} \quad i = 1, ..., n \quad . \end{aligned} \tag{6.9}$$

And the $n+3$ equations for the pedestal weights are:

$$\begin{aligned} &\textstyle\sum_{i=1}^{n} c_i g_i = 0 \quad , \\ &\textstyle\sum_{i=1}^{n} c_i g_i' = 0 \quad , \\ &\textstyle\sum_{i=1}^{n} c_i = 1 \quad , \\ &\textstyle\sum_{j=1}^{n} c_j R_{ij} - \alpha g_i - \beta g_i' - \gamma = 0 \quad \text{for} \quad i = 1, ..., n \quad , \end{aligned} \tag{6.10}$$

where λ, κ, ν, μ, ρ, ϕ, α, β and γ are the Lagrange multipliers and R_{ij} represents the element ij of the noise autocorrelation matrix defined as:

$$R_{ij} = \frac{\sum \left(n_i - \langle n_i \rangle\right)\left(n_j - \langle n_j \rangle\right)}{\sqrt{\sum \left(n_i - \langle n_i \rangle\right)^2 \sum \left(n_j - \langle n_j \rangle\right)^2}} \quad . \tag{6.11}$$

Solving the systems defined in Equation (6.8), (6.9) and (6.10) we can calculate the weights to reconstruct A, $A\tau$ and p as:

$$A = \sum_{i=1}^{n} a_i S_i \quad , \qquad A\tau = \sum_{i=1}^{n} b_i S_i \quad \text{and} \qquad p = \sum_{i=1}^{n} c_i S_i \tag{6.12}$$

The expression in Equation (6.12) is valid for the three parameter Optimal Filtering algorithm (OF2). For OF1 (two parameters) the pedestal has to be subtracted from the samples before applying the digital filter. The equations can be re-written now as:

$$A = \sum_{i=1}^{n} a_i (S_i - p) \quad , \qquad A\tau = \sum_{i=1}^{n} b_i (S_i - p) \tag{6.13}$$

In this case, the pedestal can be estimated using different methods. It is usually estimated as the average of the first and last samples, or just as the value of the first sample. It might be also possible to estimate the pedestal in dedicated pedestal runs or using zero bias events.

Moreover, we can calculate a quality factor defined as:

$$Q_{\mathrm{F}} = \sqrt{\sum_{i=1}^{n} \left(S_i - (Ag_i + A\tau g'_i + p)\right)^2} \tag{6.14}$$

which provides information about the goodness of the reconstruction. It estimates the difference between the actual samples and the ideal samples for the reconstructed amplitude and time assuming an ideal pulse shape.

6.2.1 Optimal Filtering Weights

Optimal Filtering weights are calculated from the pulse shape, its derivative and the noise autocorrelation matrix of the samples. With this information we can build up the systems from Equation (6.8), (6.9) and (6.10) and solve them to obtain the weights.

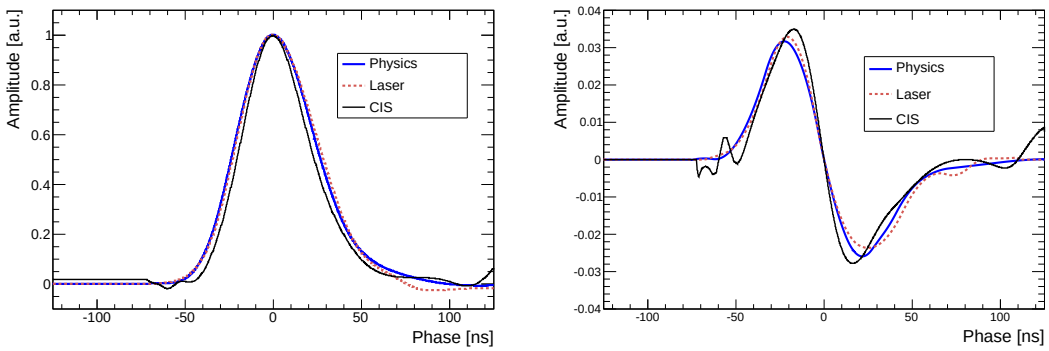


Figure 6.4: High gain pulse shape (left) and its derivative (right) for physics, CIS and laser. The pulse shapes were measured during the test beam and commissioning are stored in the conditions database.

TileCal pulse shapes and their derivatives can be found inside the conditions database in the ATLAS general software. There are three types of pulse shapes depending on the mode used to generate the signal; physics (particles crossing the scintillator), laser and charge injection (CIS). In addition, different pulse shapes are stored for high and low gains. The pulse shapes were measured during test beam and commissioning. The channel-to-channel variations were negligible (below 1%) and one pulse shape is used for all of the channels.

Figure 6.4 shows the high gain pulse shapes and their derivatives for physics (particles), CIS and laser runs. The area of the pulses for particles is 10% larger than the area of charge injection pulses.

The number of points for each pulse shape and the time range (the length of the pulse) is configurable. In the current data analysis we are using pulses for particles that are calculated between -75.5 ns and 124.5 ns in steps of 0.5 ns and pulses for charge injection and laser that are calculated between -76 ns and 122 ns in steps of 2 ns. The time origin (0 ns) is chosen to be at the pulse maximum for the pulse shape and when the derivative crosses zero for the derivative pulse. For a given initial phase τ_0, the $g(t_i)$ and $g'(t_i)$ are extracted from this data using a linear interpolation, and values out of the time range are set to the nearest value,

$$t_i = t'_i + \tau_0 = 25(i_c - i) + \tau_0 \quad \text{for} \quad i = 1, ..., n \ , \tag{6.15}$$

where i_c is the position of the central sample.

To complete the calculation of the Optimal Filtering weights the noise autocorrelation matrix is also needed. However, for test beam and commissioning the noise contribution was mainly due to the TileCal electronics. It was observed that TileCal samples were weakly correlated and the best choice in this case was to use the unitary matrix [46].

Moreover, during the first three years of ATLAS operation different studies showed that the usage of a non-unitary autocorrelation matrix had a negligible impact on the reconstruction results. This is caused by lower peak luminosity and pileup and a longer bunch spacing (50 ns) which are below the LHC design parameters. The studies will be repeated after Long Shutdown 1 (2013-2014) where the luminosity and pileup will increase and the bunch spacing might be reduced to the design value of 25 ns.

Figure 6.5 shows the Optimal Filtering weights for OF2 used to calculate the (a) amplitude, (b) phase and (c) pedestal of the signal as function of the initial phase τ_0. The Optimal Filtering weights for the amplitude follow a similar tendency as the pulse shape values. In this way the largest weights are applied to the largest samples, and the samples with low signal contribution are weighted by the smallest Optimal Filtering weights. The Optimal Filtering weights for the phase calculation follow the tendency of the derivative of the pulse, changing the sign of the Optimal Filtering weights in the two sides of the peak. Finally, the Optimal Filtering weights to calculate the pedestal follow the inverse distribution of the pulse shape giving more importance to the samples far from the peak.

The weights are calculated and stored in the conditions database from -100 ns to +100 ns in steps of 0.1 ns. In addition, the digitizing clock used in the front end electronics is tuned to have the fourth sample close to the peak of the pulse. The residual time between the fourth sample and the peak is calibrated using real data and stored in the conditions database for each channel. Ideally, any particle coming from the interaction point should have a reconstructed time equal to zero. The weights used for each channel are selected according to this expected phase stored in the database. Figure 6.6 shows the 7 weights for a expected phase of 0 ns, -10 ns and +10 ns.

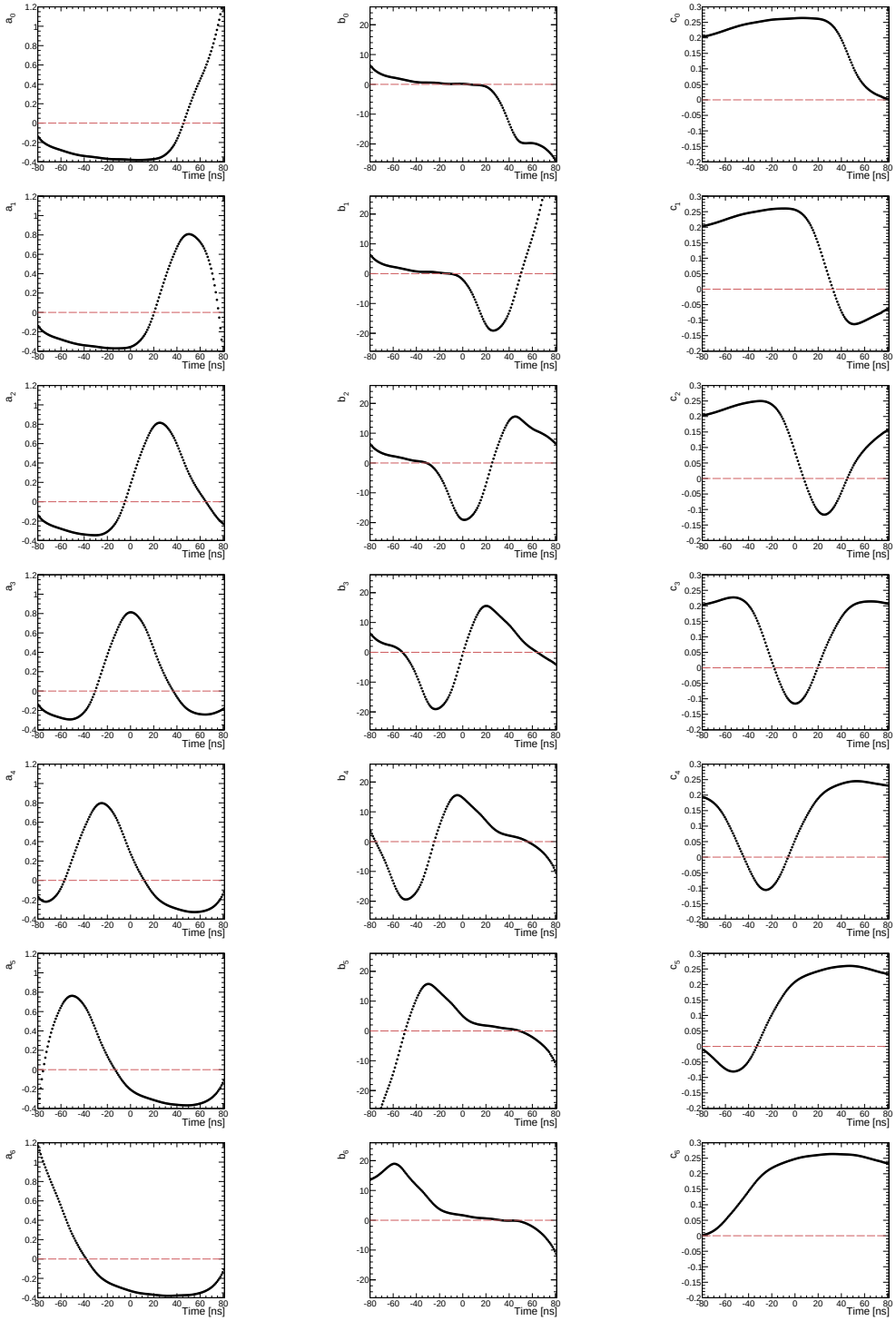


Figure 6.5: Optimal Filtering weights for the amplitude (left), phase (middle) and pedestal (right) calculation as a function of time.

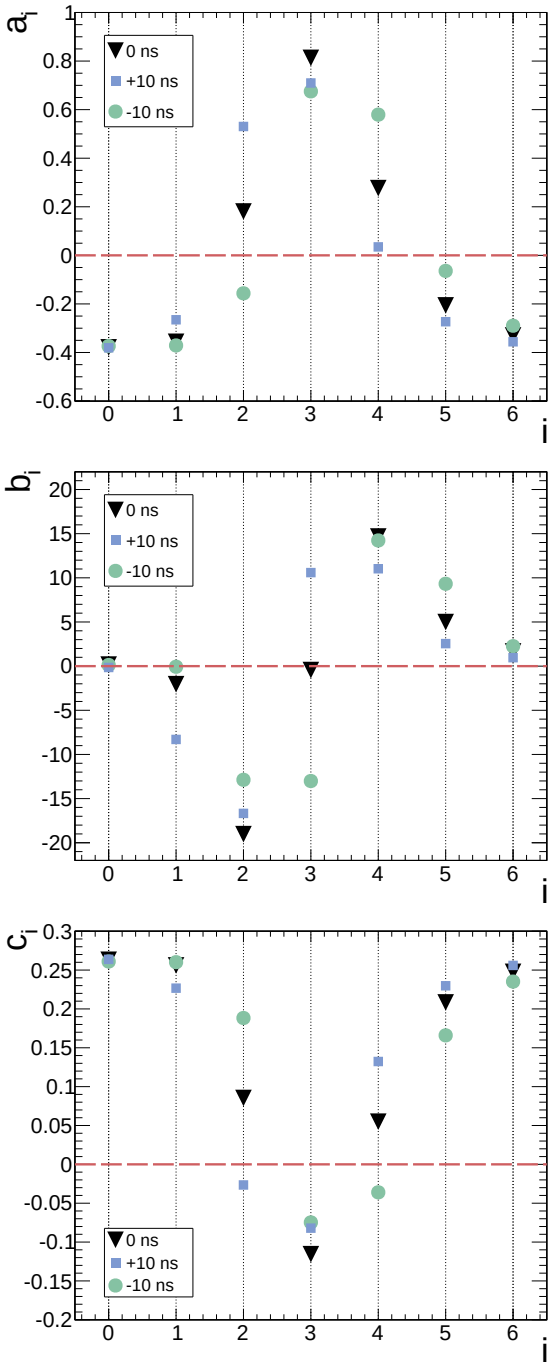


Figure 6.6: The seven Optimal Filtering weights for amplitude (top), phase (middle) and pedestal (bottom) for expected phase of 0 ns (black), +10 ns (blue) and -10 ns (green).

6.2.2 Iterations Method for Cosmic Ray Muons

The Optimal Filtering algorithm provides a good estimation of the amplitude, time and pedestal for pulses with fixed or small phase variations which occurs for particles generated at the LHC collisions. However, it does not occur in cosmic runs, with signals produced by cosmic muons crossing the detector asynchronously. In this case, Optimal Filtering can be applied and it is possible to obtain an accurate reconstruction incorporating an iteration procedure which increases the processing time but improves the reconstruction. Nevertheless, the iterative algorithm can be used for online reconstruction without introducing dead-time if the Level 1 trigger rate is kept below 10 kHz, which is the case during cosmic muon runs. The iterative method was also used during the first LHC splash events and collisions with a low Level 1 trigger rate.

The iteration procedure consists of a first estimation of the initial phase that ascertains the index of the maximum sample. If the maximum sample is the central sample, the first iteration begins with Optimal Filtering weights calculated for:

$$\tau_0 = 0 \text{ ns} \tag{6.16}$$

while, if the maximum sample differs from the central sample, the weights are calculated for:

$$\tau_0 = 25(i_c - i_{max}) \text{ ns} \tag{6.17}$$

with i_{max} and i_c the index of the maximum sample and the central sample, respectively.

Afterward the amplitude, phase and pedestal are calculated in each iteration as:

$$A_k = \sum_{i=1}^{n} a_i \Big|_{\tau_{k-1}} S_i \tag{6.18}$$

$$\tau_k = \frac{1}{A_k} \sum_{i=1}^{n} b_i \Big|_{\tau_{k-1}} S_i \tag{6.19}$$

$$p_k = \sum_{i=1}^{n} c_i \Big|_{\tau_{k-1}} S_i \tag{6.20}$$

where k is the iteration index starting from 1. It has been observed that the amplitude converges within three iterations for Optimal Filtering weights calculated in steps of 1 ns, which is the number of iterations executed in the DSPs if this method is selected.

6.2.3 Energy Calibration

The amplitude of the pulse reconstructed in the DSP using the Optimal Filtering is estimated in ADC-counts. However, due to the different response of the channels the amplitude is calibrated to units of energy. The DSP can provide the amplitude in ADC-counts, MeV or pC depending on the run type. The reconstructed channel energy used by the HLT and offline in physics is provided by the DSP in units of MeV units calibrated as:

$$E_{channel} = A \times K_{ADC \to pC} \times K_{pC \to GeV} \times K_{Cs} \times K_{Laser} \tag{6.21}$$

The signal amplitude A represents the reconstructed amplitude in ADC-counts as in Equation (6.12). The $K_{ADC \to pC}$ is the conversion factor of ADC to charge and it is determined using a well defined injected charge with the CIS calibration system. The $K_{pC \to GeV}$ is the conversion factor of charge to energy in GeV and it has been defined at testbeam for a subset of modules via the response to electron beams of known momentum. This is a global factor and has a layer dependence. The K_{Cs} corrects for residual non-uniformities after the gain equalization of all channels has been performed by the Cs radioactive source system. The K_{Laser} corrects for non-linearities of the PMT response measured by the Laser calibration system. The derived time dependence of the last two factors will be applied to preserve the energy scale of TileCal. A unique calibration constant to convert ADC-counts to MeV is retrieved from the conditions database for each channel and copied in the DSP memory at the start of each run.

6.3 The DSP Reconstruction Algorithms

Each TileCal ROD has four Texas Instruments TMS320C6414 DSPs which provide an instruction cycle frequency of 720 MHz, 1024 kB of user memory and an additional 32 kB level

1 cache memory. It has eight highly independent functional units divided in six Arithmetic Logic Units (ALU), supporting single 32-bit, dual 16-bit or quad 8-bit arithmetic per clock cycle and two multipliers supporting four 16x16-bit or eight 8x8-bit multipliers per clock cycle. This CPU structure provides a suitable framework for any digital filter because it enhances MAC operations. Every event received in the DSP is synchronized with the TTC information and is then processed. The first step reconstructs the energy and phase using the Optimal Filtering algorithm for all the channels (Figure 6.7). The energy is then calibrated and the total energy sum algorithm executed.

The next step depends on the trigger type. For the special low rate ZeroBias trigger type the digits are packed and the quality factor computed for all channels. For any other trigger type the channel signals are separated and stored in three groups. The first group contains the signals with a difference between the maximum and minimum sample lower than 16 ADC-counts in high gain but higher than a programmable threshold, which by default is 5 ADC-counts (m in Figure 6.7). The second group includes the signals with that difference higher than 15 ADC-counts or any amplitude different than zero in low gain (k in Figure 6.7). The third group contains channels with signals below the programmable threshold. In the next step two separated loops are executed for the first two groups of channels. The number of iterations of these two loops varies from event to event since it depends on the number of signals included in each group. In the first loop, the digits of signals in the first group are packed in the 6 bytes data format. In the second loop, the digits of channels in the second group are packed using the 10 bytes data format and the quality factor is computed. The digits of channels included in the third group are not transmitted and the DSP energy and time reconstruction is used for offline analysis. Thus, the quality factor is not computed for channels with small signals and it is forced to zero, which is in any case the reconstructed quality factor for these types of signals.

6.3.1 Optimal Filtering Implementation for LHC Operation

The Optimal Filtering magnitudes computed in the DSP are given in Equation (6.22). Note that even if the OF2 method is used, the pedestal is not computed in the DSP since it is not transmitted. The pedestal used for the QF calculation is estimated as the first sample.

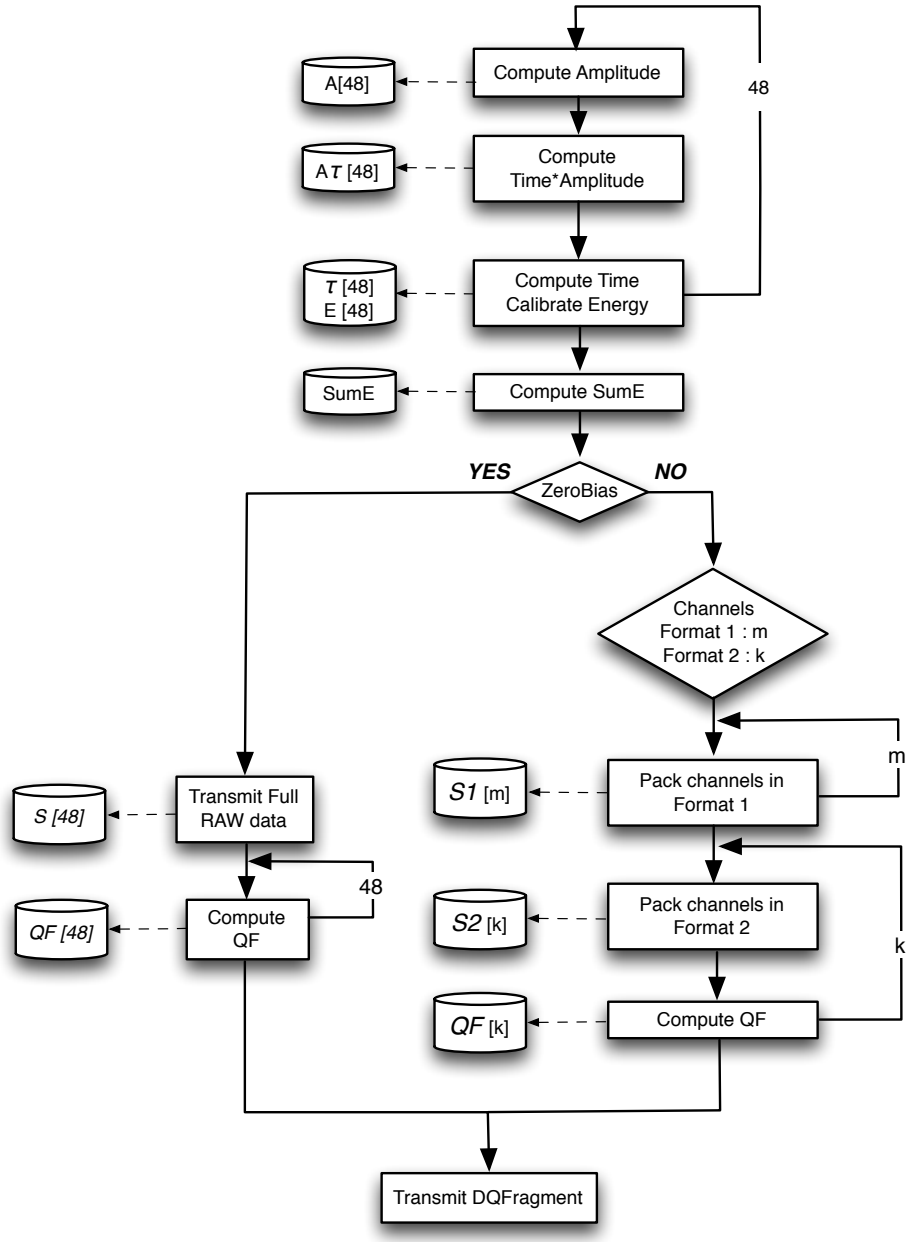


Figure 6.7: DSP reconstruction algorithm code flow.

32-bit words		
10 bits	10 bits	10 bits
Sample1 CH3	Sample1 CH2	Sample1 CH1
Sample2 CH3	Sample2 CH2	Sample2 CH1
Sample3 CH3	Sample3 CH2	Sample3 CH1
Sample4 CH3	Sample4 CH2	Sample4 CH1
Sample5 CH3	Sample5 CH2	Sample5 CH1
Sample6 CH3	Sample6 CH2	Sample6 CH1
Sample7 CH3	Sample7 CH2	Sample7 CH1

→

32-bit words	
10 bits	10 bits
Sample1 CH1	BCID
Sample3 CH1	Sample2 CH1
Sample5 CH1	Sample4 CH1
Sample7 CH1	Sample6 CH1
Sample1 CH2	BCID
Sample3 CH2	Sample2 CH2
Sample5 CH2	Sample4 CH2
Sample7 CH2	Sample6 CH2
Sample1 CH3	BCID
Sample3 CH3	Sample2 CH3
Sample5 CH3	Sample4 CH3
Sample7 CH3	Sample6 CH3

Figure 6.8: DSP input data format adaptation.

$$A = \sum_{i=1}^{7} S_i a_i \qquad \tau = \frac{1}{A} \sum_{i=1}^{7} S_i b_i \qquad QF = \sqrt{\frac{1}{512} \sum_{i=1}^{7} S_i - A g_i + A \tau g'_i - ped} \tag{6.22}$$

The weights are downloaded in the DSP memory through the Host Port Interface (HPI) at configuration time. A set of Optimal Filtering weights, including calibration constants, for each channel and gain are stored. The weights for each channel are selected according to the expected phase which is computed with offline software and stored in the ATLAS conditions database. The format of the weights in the DSP memory is shown in Table 6.1. The storage of the weights and calibration constants in the DSP memory space has been organized to optimize the operations in the reconstruction algorithms.

The front-end data format is reorganized in order to reduce the processing time in the DSP.

The DMU-block structure is replaced by a channel structure (Figure 6.8). In order to optimize the processing time in the DSP the formatting of the input data is performed in the InputFPGA before its transmission to the DSP. The format of weights (Table 6.1) and samples (Figure 6.8) has been chosen to improve the parallel multiplications of samples and

Nb. word	32-bit words	
	16 bits	16 bits
0	a_1	Scale a
1	a_2	a_3
2	a_4	a_5
3	a_6	a_7
4	Calib offset	Calib Const
5	b_1	Scale b
6	b_2	b_3
7	b_4	b_5
8	b_6	b_7
9	p_1	Scale p
10	p_2	p_3
11	p_4	p_5
12	p_6	p_7
13	Scale g'	Scale g
14	g'_1	g_1
15	g'_2	g_2
16	g'_3	g_3
17	g'_4	g_4
18	g'_5	g_5
19	g'_6	g_6
20	g'_7	g_7
21	Channel ID	Phase COOL

Table 6.1: Description of the Optimal Filtering weights and calibration constants for one channel and gain in the DSP memory.

weights. Figure 6.10 shows a graphical representation of the *dotp2* intrinsic function which is executed in the DSP in four clock cycles.

The first routine executed within the processing task of the DSP computes the amplitude and phase of the pulse (Figure 6.9). The amplitude of the signal is estimated as a multiplication of samples and Optimal Filtering weights using the *dotp2* intrinsic function (Figure 6.10). The result is the amplitude of the signal in ADC-counts multiplied by the scaling factor (Scale_a) of Optimal Filtering weights:

$$\sum_{i=1}^{7} 2^{Scale\_a}\, a_i\, S_i = A[\mathrm{ADC}]\, 2^{Scale\_a} \tag{6.23}$$

The 10-bit samples and the signed 16-bit weights lead to a maximum possible result of 28-bit.

Then, the result is divided by a factor before the multiplication by the 16-bit calibration constant (K) in order not to overflow the 32-bit register:

$$\frac{A[\text{ADC}]\,2^{Scale\_a}}{2^{11}} \longrightarrow A'[\text{Units}] = A[\text{ADC}]\,2^{Scale\_a-11}\,K\left[\frac{Units}{\text{ADC}}\right]\,2^{Scale\_K} \tag{6.24}$$

A division by 2^{11} assures a maximum calibrated amplitude of 32-bit. At this point the precision of A is 0.0625 ADCs.

Then, the a weights and calibration constant scaling factors are removed taking to account for the previous de-scaling not to overflow the 32-bit register:

$$A[\text{Units}] = \frac{A'[\text{Units}]}{2^{Scale\_a+Scale\_K-11}} \tag{6.25}$$

Finally, the result is saturated and packed in the 15-bit word reserved in the output bytestream (Table 6.2).

The first step in the phase computation is the multiplication of weights and samples. Again, the weights are scaled by a factor to improve the precision in the fixed point arithmetic used in the DSP:

$$\sum_{i=1}^{7} 2^{Scale\_b}\,b_i\,S_i = \tau\,A\,2^{Scale\_b} \tag{6.26}$$

Both, $2A$ (in ADC-counts) and $A\tau$ are stored in the memory and later used in the quality factor computation.

The division by the energy needed to obtain the phase of the pulse is performed using a Look Up Table (LUT). The output of the LUT is $\frac{1}{A}$ multiplied by a scaling factor. The scaling factor used for the LUT is 2^{15}. Hence, the phase is obtained by a multiplication of Equation (6.26) and the output of the LUT for the amplitude previously reconstructed in ADC-counts:

$$\tau' = \tau\,A\,2^{Scale\_b}\,\text{LUT}(A)\,2^{Scale\_\text{LUT}} \tag{6.27}$$

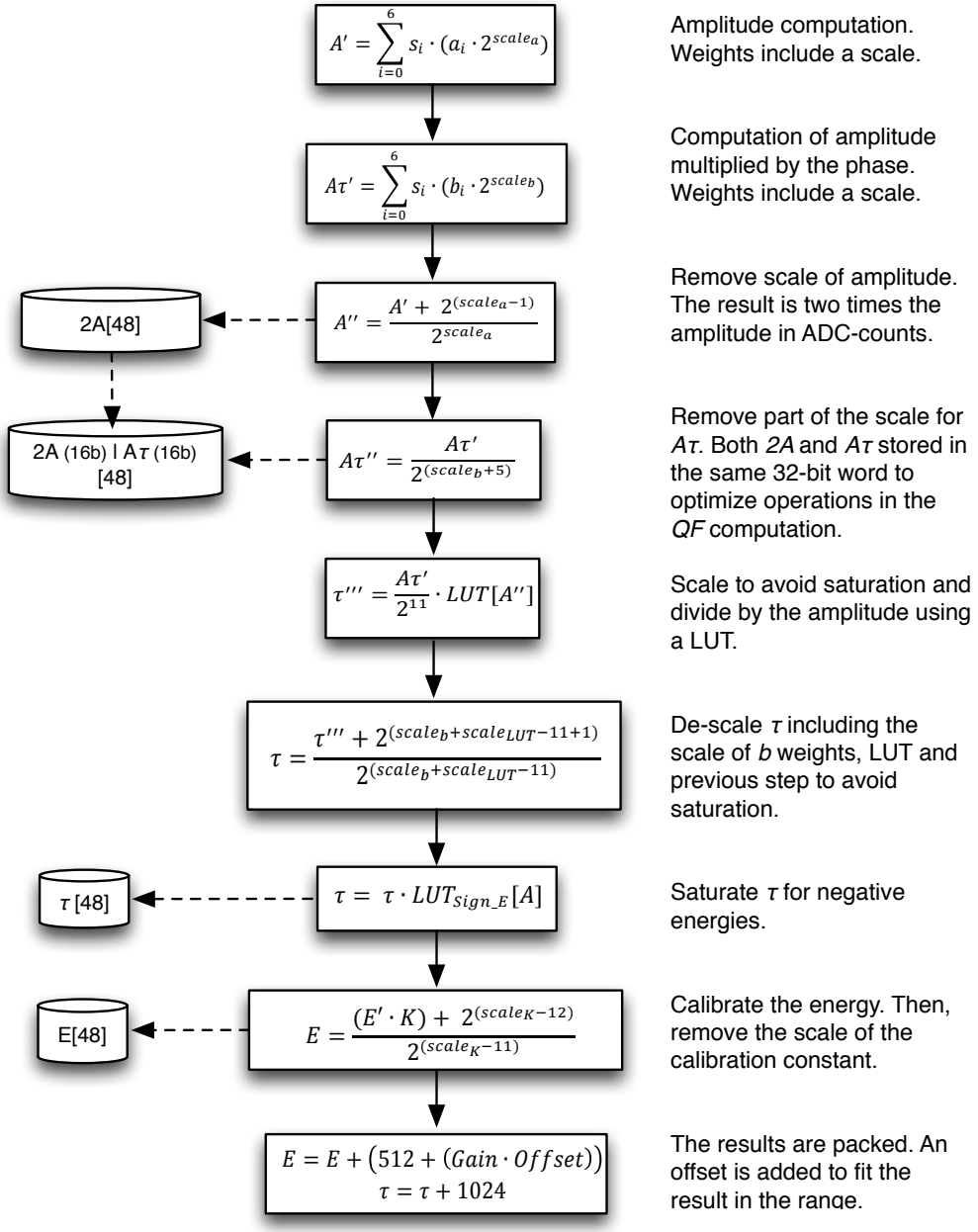


Figure 6.9: Code flow for the energy and time reconstruction in the DSP.

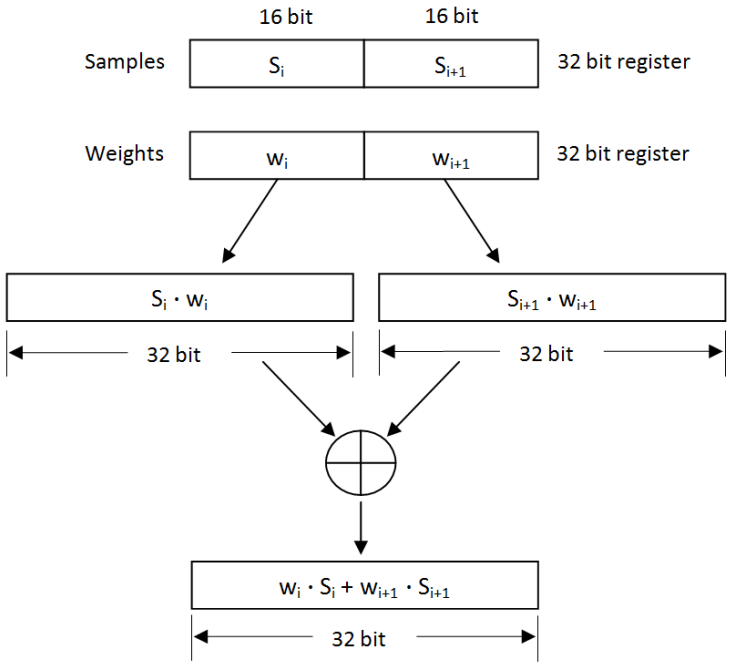


Figure 6.10: Graphical representation of the *dotp2* intrinsic function.

Then, the b weights and the LUT scaling factors are removed to obtain the phase result in ns:

$$\tau[ns] = \frac{\tau'}{2^{Scale\_b + Scale\_\mathrm{LUT}}} \tag{6.28}$$

The Quality Factor Computation

The quality factor is only computed for channels in low gain or in high gain with a difference between a maximum and minimum sample higher than 15 ADC-counts. The first step is to subtract the pedestal in all the samples (Figure 6.11).

In order to reduce the processing time the pedestal is estimated from the first sample. Then, the amplitude and phase previously estimated are multiplied by g and g' weights:

$$S'_i = S_i - S_0 \longrightarrow \sum_{i=1}^{7} S'_i - \frac{g_i\, A}{2^{Scale\_g}} + \frac{g'_i\, A\, \tau}{2^{Scale\_g'}} \tag{6.29}$$

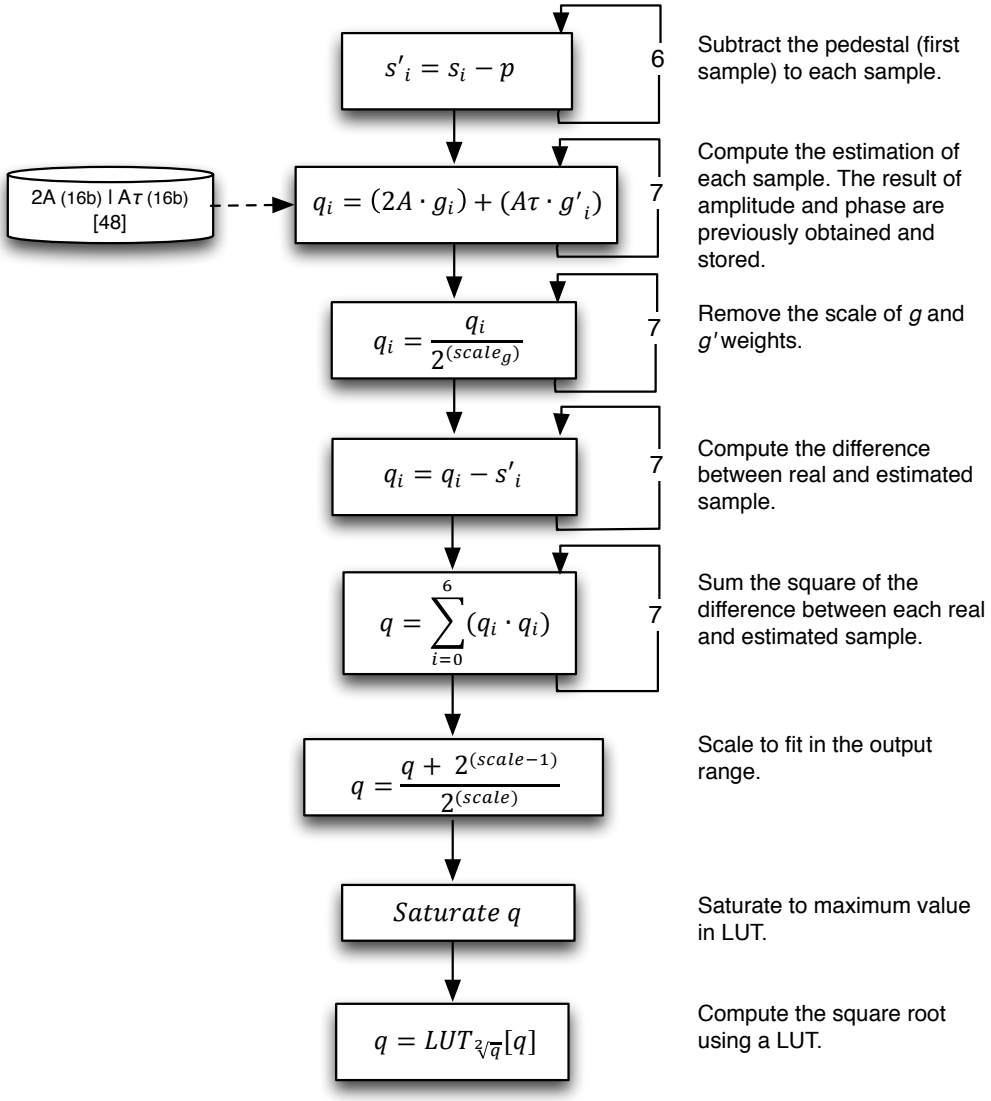


Figure 6.11: Code flow for the quality factor reconstruction in the DSP.

The scaling factors used for g and g' are removed before executing the summation term. Note that the sign used for the g' term is opposite to the g term due to the sign of the phase computed with Optimal Filtering.

Finally, the square root of the result obtained in Equation (6.29) is implemented as another LUT.

Since only 4 bits are used to pack the QF result it has to be scaled (Table 6.2). The division by a scaling factor is performed before accessing the square root LUT in order to have a smaller LUT. Then, the input of the square root LUT is the result obtained in Equation(6.29) divided by a scaling factor and saturated to the maximum possible value in the LUT. The scaling factor used before accessing the LUT is 2^9 which implies an actual scaling factor of $\sqrt{512}$. The result gives an estimation of the total deviation between received and expected samples for a given pulse shape in ADC-counts.

DSP Constraints and Expected Precision

The main constraints in the DSP computation are different for each Optimal Filtering magnitude. The expected precision of the amplitude is limited by different DSP constraints. One limitation is due to the usage of fixed point arithmetic which limits the precision to 0.0625 ADC-counts (Equation (6.24)). Then, the result is multiplied by a calibration constant which is packed in a unsigned 16-bit word. The precision of this constant is different for each channel since the scaling factor is chosen as the maximum possible value for each case as explained in Section 6.3.1. The precision of the final result depends on the precision of the amplitude computation in ADC-counts and the precision of the calibration constant:

$$A[\text{Units}] = A[\text{ADC}] \cdot K \left[\frac{\text{Units}}{\text{ADC}}\right] \longrightarrow \Delta A[\text{Units}] = \frac{\partial A[\text{ADC}]}{\partial A}\Delta A[\text{ADC}] + \frac{\partial A[\text{ADC}]}{\partial K}\Delta K \quad (6.30)$$

The expected precision for the calibrated amplitude has two terms: the first term is the precision of the amplitude in ADC-counts multiplied by the calibration constant and the second term is the precision of the calibration constant multiplied by the amplitude in ADC-counts:

$$\Delta A[\text{Units}] = K\Delta A[\text{ADC}] + A[\text{ADC}]\Delta K \quad (6.31)$$

The precision of the amplitude in ADC-counts is 0.0625 ADCs while the precision of the calibration constant depends on the fractional bits for the fixed point representation of the constant, which is different for each channel.

Finally, the data is packed in the output bytestream using 15-bits. This packaging of the result restricts the range and the precision depending on the calibration units as shown in Table 6.2.

The phase is estimated in the DSP dividing the $A\tau$ product by the obtained amplitude. The $A\tau$ term is computed using the samples and b weights whereas the division by the energy is performed using a LUT. The uncertainty of the phase computed in the DSP can be written as:

$$\tau = \mathrm{LUT}(A)\underbrace{\sum_{i=1}^{7} b_i S_i}_{A\,\tau} \longrightarrow \Delta\tau = \frac{\partial\tau}{\partial\mathrm{LUT}(A)}\Delta\mathrm{LUT}(A) + \sum_{i=1}^{7}\frac{\partial\tau}{\partial S_i}\Delta S_i + \sum_{i=1}^{7}\frac{\partial\tau}{\partial b_i}\Delta b_i \qquad (6.32)$$

$$\Delta\tau = A\,\tau\Delta\mathrm{LUT}(A) + \mathrm{LUT}(A)\sum_{i=1}^{7} b_i + \mathrm{LUT}(A)\sum_{i=1}^{7} S_i\Delta b_i \qquad (6.33)$$

The summation of b weights in OF2 is zero and the second term in Equation 6.33 is removed. In addition, the third term depends on the uncertainty of the b weights which is different for each channel. However, the scaling factor of the b weights is chosen to introduce a maximum uncertainty of 0.1 ns in the phase result. Hence, the main constraint in the phase computation is introduced by the usage of the LUT. The uncertainty of the LUT depends on the value of the energy. Figure 6.12 shows the difference between exact value of $\frac{1}{E}$ and the corresponding value in the LUT. In addition, the phase uncertainty introduced by the LUT increases with the phase and the amplitude of the pulse Equation (6.33). Then, the expected precision in the phase computed in the DSP depends on the phase, amplitude and the uncertainty in the LUT.

Finally, the limitation due to the number of bits available to pack the result is 0.0625 ns and the range of phases varies from -64 ns to 64 ns. The DSP phase result is saturated if it exceeds this range.

31	30.....16	15.....5	4	3.....0
Gain	Energy	Phase	HLT	QF

Table 6.2: Data format for the reconstruction word of a TileCal channel.

Energy Units	**Emin**	**Emax**	**Precision**
Online ADC	-32	2016	0.0625
Online Pico Coulombs	-16	1008	0.03125
Online Mega Electron Volt	-16384	1032160	32

Table 6.3: Range and precision of the energy for the different energy units for LG.

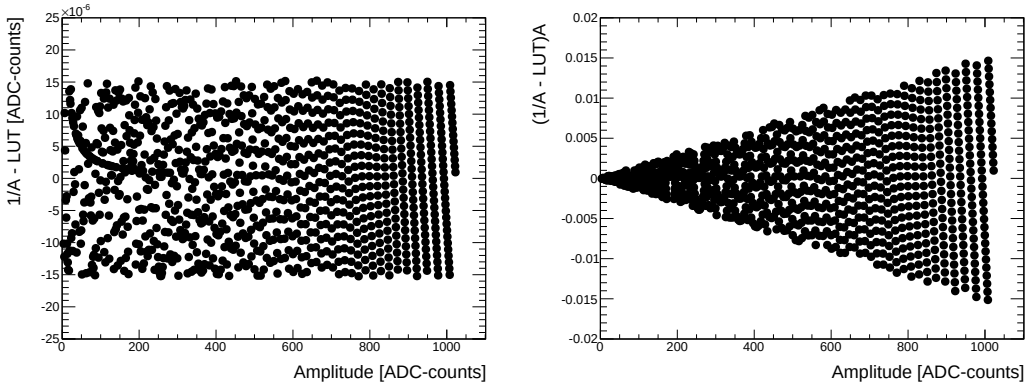


Figure 6.12: Absolute (left) and relative (right) difference between 1/E and the corresponding value in the LUT as a function of the amplitude.

Optimal Filtering Weights and Calibration Constants Plugin

Condition constants needed for DSP processing are stored in the ATLAS COOL conditions database [47] in floating point precision. These are retrieved by the online software at con-

Energy Units	**Emin**	**Emax**	**Precision**
Online ADC	-128	1920	0.0625
Online Pico Coulombs	-1	15	0.00048828125
Online Mega Electron Volt	-1024	15360	0.5

Table 6.4: Range and precision of the energy for the different energy units for HG.

figuration time in the *prepare for run* transition, packed and transferred into the DSPs with a limited precision. The condition data required online which are stored in COOL are the Optimal Filtering weights for a generic pulse shape for 1500 different phases (0.1 ns step), which contain the a, b, c constants, the normalized pulse shape value (g) and the derivative of the pulse shape value (g'), the calibration constants per channel and gain from ADC to pC ($\mathrm{K}\left[\frac{\mathrm{pC}}{\mathrm{ADC}}\right]$), the equalization constants per channel and gain for Cs (e[Cs]) and Laser (e[Laser]), the calibration constants per channel from pC to MeV ($\mathrm{K}\left[\frac{\mathrm{MeV}}{\mathrm{pC}}\right]$), the channel status per gain, the muon identification thresholds per drawer and gain, and the conditions to dump the samples in the compressed data fragment per drawer and gain.

The RCD application running in each TileCal ROD crate manages the transmission of the constants to the DSP internal memory. Despite the quantity of different types of condition constants, there are only four types of fragments that are transferred to the DSP. The Optimal Filtering Constants (OFC) contain the Optimal Filtering weights (a, b, c, g, g') and the final calibration constants, which are computed like the following for Cesium, equalized pC (CspC) and MeV:

- $\mathrm{K}\left[\frac{\mathrm{CspC}}{\mathrm{ADC}}\right] = \mathrm{K}\left[\frac{\mathrm{pC}}{\mathrm{ADC}}\right] \times \mathrm{e[Laser]} \times \mathrm{e[Cs]}$

- $\mathrm{K}\left[\frac{\mathrm{MeV}}{\mathrm{ADC}}\right] = \mathrm{K}\left[\frac{\mathrm{pC}}{\mathrm{ADC}}\right] \times \mathrm{e[Laser]} \times \mathrm{e[Cs]} \times \mathrm{K}\left[\frac{\mathrm{MeV}}{\mathrm{pC}}\right]$

The packed conditions are read out by the RCD and stored in the shared memory of the ROD crate controller. This is done at the *prepare for run* stage, thus, only the required constants are read out. Neither the constants for non powered drawers nor the conditions for different run types to the given one are read out. Following that, each ROD module reads the packed constants stored in the shared memory and writes them to the DSPs through the HPI.

The packing of the OFC is such that the algorithm inside the DSP can easily handle iterations and no-iterations, thus there are two types of packing. Either way the value of each weight (a,b,c,g,g') is packed as a 16-bit integer value and a scale which is the same for the same channel.

The scale is obtained from the highest absolute value of the weights or as the sum of the

weights which we call max, using the following equation:

$$\text{scale} = \text{truncf}\left(\frac{\frac{log2^{15}-1}{\max}}{log2}\right) \tag{6.34}$$

The value is obtained as the product of the floating point value times two to the power of the scale:

$$\text{value} = \text{floating point value} \times 2^{\text{scale}} \tag{6.35}$$

The value of the calibration product is also packed into a 16-bit integer value and scale using the same expressions where in this case, max is just the value of the calibration product.

6.3.2 Parabolic Deviation and Offline Correction

The Optimal Filtering weights are computed for each channel according to the expected phase of a pulse produced by a particle coming from the interaction point. The reconstructed energy is underestimated for any phase variation and the correct result is only obtained for pulses with reconstructed phase equal to zero.

Figure 6.13 shows the difference between the Optimal Filtering and the iterative algorithm as a function of the pulse phase. The iterative method selects a different set of weights according to the phase of each pulse and therefore is not affected by phase variations. The two methods are equivalent for pulses with phase equal to zero (blue points) and the difference increases with the absolute value of the phase. This feature reduces the effect of out of time Minimum Bias pileup noise because the energy of pulses from different bunch crossing is underestimated with respect to depositions with small phases.

However, small phase variations are expected for particles produced in the bunch crossing of interest due to particles with different momentum or crossing the cell through different paths. The bias produced in the energy by phase variations can be parametrized to correct the deviation offline using the reconstructed phase. The red points in Figure 6.13 shows the difference between the iterative and the standard optimal filtering methods if the so-called parabolic correction is applied to the second one. The difference between the two methods is reduced to less than 5% in the time range [-25,25] ns.

The parabolic correction is not applied to pulses with large phases to reduce the effect

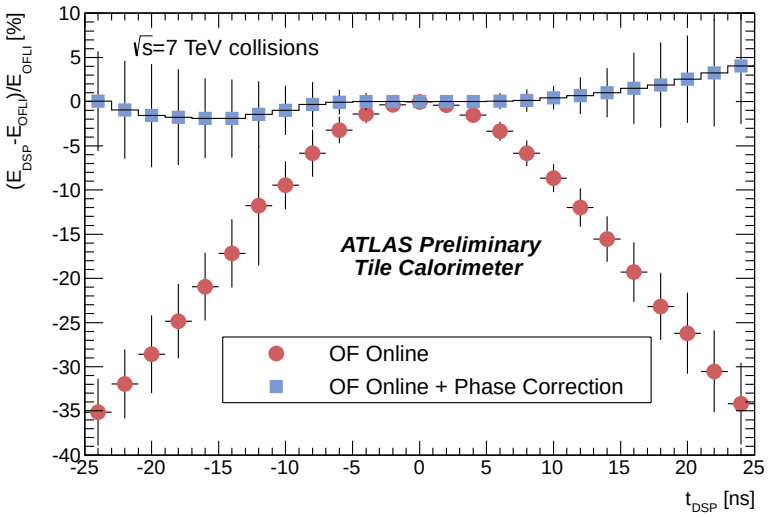


Figure 6.13: Relative difference between the Optimal Filtering iterative and non-iterative algorithms without (red) and with (blue) parabolic correction applied as a function of the reconstructed phase.

of of the out-of-time pileup. Pulses with a phase higher than half of the bunch spacing are considered out-of-time pileup. Therefore, the parabolic correction is applied in a time range defined by the bunch spacing. For the 50 ns bunch spacing configuration used in the three first years of ATLAS operation, the parabolic correction was applied in the time range [-25,25] ns.

6.3.3 Total Energy Sum Algorithm for the Level 2 E_T^{miss} Trigger

The total energy sum algorithm of the DSP computes the total energy of each module and its transverse and Z projections. The result is packed in a dedicated fragment which is used in the HLT to implement a E_T^{miss} trigger. The information provided by the DSP reduces the processing time and the flow of data in the HLT system. The algorithm is executed for each module after the reconstruction of the channels energy which is used to compute the total energy sum. The fragment is only executed if no digital errors are found in the DQ fragment in order to avoid the processing of corrupted data that might produce fake E_T^{miss} at HLT. First, channels with energy higher than 3σ of the noise are selected (Figure 6.14). The 3σ of the

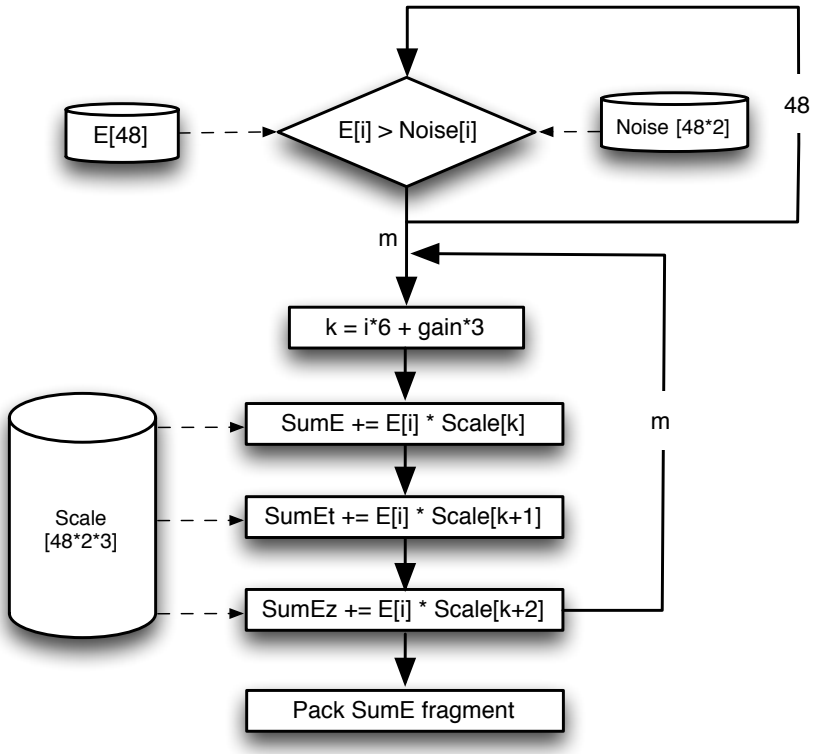


Figure 6.14: Code flow of the total energy sum algorithm.

noise is stored in the DSP memory at configuration time for each channel and gain. Then, a loop with variable number of iterations is executed for the selected channels. The total energy sum and the two projections are computed as a sum of the channel energy multiplied by three different scales. The scale for the total energy sum includes a factor 64 for channels in low gain and a factor of 2 for channels connected to a cell with the second PMT masked as bad in the conditions database. The scale of channels masked as bad is zero and does not contribute to the energy sum. The scale for the transverse projection includes the previous factors multiplied by $sech(\eta)$ of the corresponding channel. Finally, the scale for Z projection includes the factors of the first scale multiplied by $cos\phi$ which is module dependent. The three scales are computed once at configuration time and stored in the internal DSP memory.

Chapter 7

Validation and Performance of the DSP Optimal Filtering

7.1 Introduction

During the commissioning phase the data processed by the RODs was used for the calibration systems (CIS and Laser) and cosmic muons runs. The absence of pileup and the low Level 1 Trigger rates made the Optimal Filtering iterative method the most appropriate algorithm for the data reconstruction both online and offline, and was the default reconstruction method during this period. In 2010 the iterative method was also used as the default reconstruction algorithm in TileCal. The first data acquired from collisions was used to calibrate the TileCal channel timing, which is essential to have good performance using the non-iterative Optimal Filtering algorithm. Then, the non-iterative Optimal Filtering became the default reconstruction method both online and offline to minimize the effect of the out-of-time pileup and to operate the RODs at high Level 1 Trigger rates, minimizing the deadtime.

This Chapter describes the validation performed to assure a proper implementation of the Optimal Filtering algorithm in the fixed point arithmetic of the DSPs. The procedure used to calibrate the detector timing, which is crucial for a proper signal reconstruction, is also presented. Then, the performance of the reconstruction in the DSP is studied first under

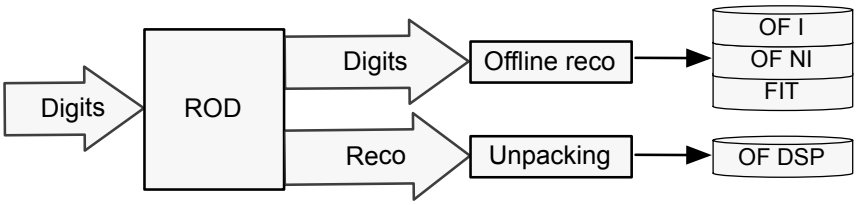


Figure 7.1: Data flow to validate the reconstruction in the DSP.

controlled conditions and second using real collision data. Finally, the impact of the limited precision of the DSP arithmetic in jet reconstruction is analyzed.

7.2 Validation of the DSP Optimal Filtering Implementation

The ATLAS offline reconstruction software (Athena) includes different signal reconstruction algorithms for TileCal. If the samples are available, it is possible to use Optimal Filtering (with and without iterations) or the so-called Fit method to reconstruct offline the digits (Figure 7.1). This software uses floating point arithmetic and is usually used as a reference to analyze and quantify the goodness of the online reconstruction. The fixed point arithmetic of the DSPs provides a limited precision in the reconstruction which was analyzed and quantified in the previous Chapter. In order to validate the precision estimations and to assure the correct implementation of the Optimal Filtering method in the DSP, the same algorithm was emulated in the Athena framework using floating point arithmetic which can be considered to have unlimited precision operations. It consists of a modified version of the offline Optimal Filtering non-iterative algorithm but limiting the precision of every operation to emulate the fixed point arithmetic.

Figure 7.2 shows how both online and offline with limited precision produce exactly the same result for the energy (left) and time (right) reconstruction. This result was obtained for a scan in all the energy range in a CIS run. The reconstruction was configured not to apply any calibration constants and the amplitude of the pulse was obtained in ADC-counts to

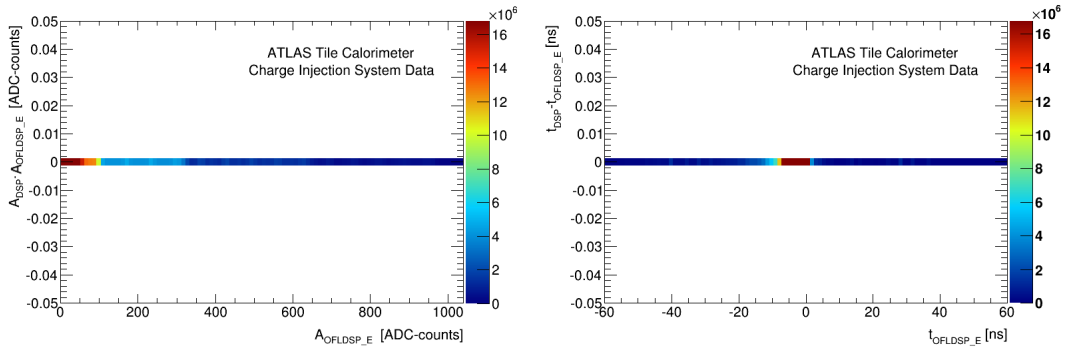


Figure 7.2: Absolute difference in the amplitude (left) and time (right) reconstructed by DSP and offline (with limited precision) for a CIS run as a function of the amplitude (left) and time (right) reconstructed offline.

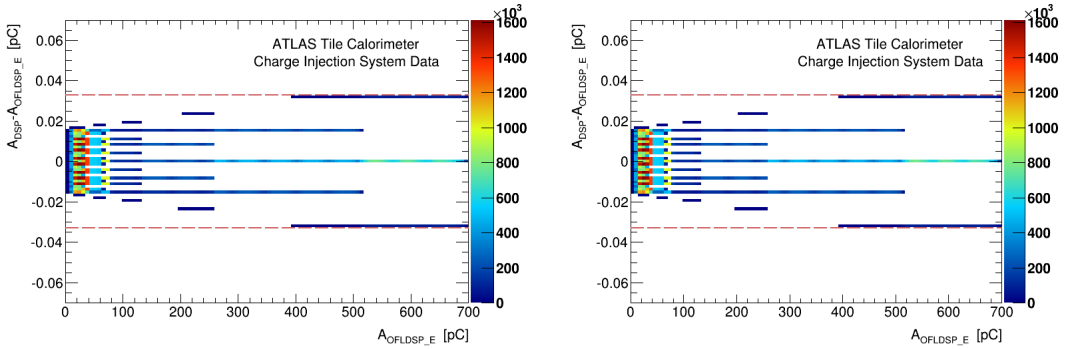


Figure 7.3: Absolute difference in the energy calibrated in pC units reconstructed by DSP and offline with limited precision for a CIS run as a function of the amplitude reconstructed offline in HG (left) and LG (right). The residual differences are produced by a non-linear term only applied in the calibration constant used offline.

avoid differences produced by the different precision between the online and offline calibration constants. These results validate a correct implementation of the Optimal Filtering algorithm in the DSP with the expected limitation of the fixed point arithmetic.

The effect of the different precision of the calibration constants online and offline can be seen in Figure 7.3. In this case, the offline amplitude reconstruction also emulates the DSP precision but the calibration from ADC-counts to pC units is applied with a different precision than is used in online and offline. The difference between online and offline calibration is

produced by two factors. First, the calibration constant is a 16-bit value in the DSP and a 64-bit (double) in the offline case. The second factor comes from rounding and packing of the result in the DSP. Table 3.24 shows the range and precision of the DSP result for each calibration unit. Essentially, the precision of the DSP is defined by the value of the Least Significant Bit (LSB) of the packed result. In Figure 7.3 we observe a quantized difference between DSP and offline result caused by the floating point arithmetic used offline as a comparison with the DSP fixed point arithmetic. In any case, the difference between the methods is smaller than the minimum precision of the DSP result, represented by the dashed red lines in the plots.

7.3 Qualification of DSP Reconstruction Under Controlled Conditions

In Chapter 4 it was shown how the OMB can be used to emulate digitized PMT pulses which are injected into the ROD to emulate the front-end electronics data. It is possible to configure the amplitude, time and a flat pedestal for the injected pulses which are generated by the OMB using an ideal free-of-noise physics pulse. This operation mode allows the qualification of the DSP reconstruction as a comparison with the true injected amplitude and time. On the other hand, the CIS provides the possibility of injecting configurable analog signals in the base of the PMTs which are afterwards digitized and transmitted to the RODs. In this case, the pulses generated also include variations mainly due to electronic noise. However, it provides a system to validate the DSP Optimal Filtering under controlled real conditions with the real electronics system.

7.3.1 DSP Reconstruction with Pseudo-Data

The OMB injection mode was used to study the performance of the DSP reconstruction with an ideal pulse shape. Table 7.1 shows the parameters of the pulses injected into the ROD which essentially covered the entire amplitude and phase ranges in both gains. Only the low gain pulse shape is available in the OMB to generate the samples both for high and low gains.

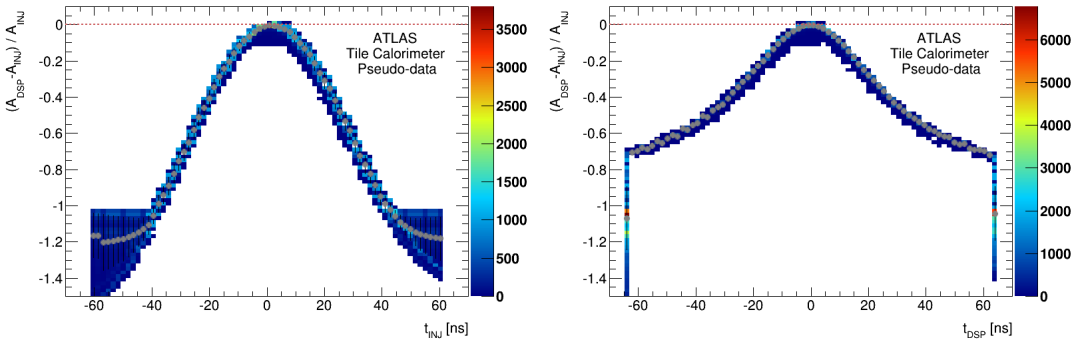
Amplitude range	[0,999] ADC-counts
Amplitude step	1 ADC-count
Time range	[-60,60] ns
Time step	1 ns
Gain	HG and LG
Total events	242000

Table 7.1: Parameters of the pseudo-data run from the OMB operating in injection mode.

However, the weights used in the DSP to reconstruct high and low gain pulses are different. Therefore, it is expected to obtain different reconstruction results for each gain even if the differences in the pulse shape are minimum as we have shown in the previous Chapter.

A wide time range was selected in order to study the bias produced in the reconstruction by the time variations in the pulse. Figure 7.4a shows the relative difference between the amplitude reconstructed by the DSP and the actual injected amplitude as a function of the injected phase for one channel. The difference is minimum for pulses with phase equal to the expected phase which is 2 ns for the channel used in this example. The relative difference increases with the time of the pulses and is larger than 100% for pulses with absolute reconstructed time above approximately 40 ns. The large spread for these negative amplitudes is produced by the reduced range reserved for negative amplitudes in the data format (Table 3.24). Thus, by definition the DSP will reconstruct negative amplitudes for pulses with large time deviations. The Optimal Filtering assumes that the pulse is located in the central samples and the pedestal in the two first or last samples. Therefore, a pulse with a large time deviation produces a high pedestal with a negative pulse in the central samples. This feature minimizes the effect of the out-of-time pileup but on the other hand it produces a bias in the in-time pulse amplitude reconstruction.

The so-called parabolic deviation can be parametrized as a function of the reconstructed phase in order to correct the bias produced by phase variations. Figure 7.4b shows the relative difference as a function of the reconstructed phase. The minimum difference between injected and reconstructed amplitude is obtained when the reconstructed time is zero. The parametrization was performed using physics events with different times and can be used to correct the bias in the entire phase range. In physics events, it is only applied to pulses with



(a) (b)

Figure 7.4: Relative difference between injected and reconstructed amplitude as a function of the injected phase (a) and reconstructed phase (b).

an absolute reconstruct time below half of the minimum bunch spacing, since it is assumed that pulses with larger times were produced in a different bunch crossing and therefore should not be considered for the triggered event. Nevertheless, Figure 7.5 shows that the bias can be reduced to less than 5% in the entire range.

The correlation between the injected and reconstructed amplitudes for pulses with time within one bunch crossing (±12.5 ns) after applying the parabolic correction can be seen in Figure 7.6 for high and low gains. The reconstructed amplitudes show a good linearity in the entire range for both gains.

The residual is calculated as the absolute difference in ADC-counts between injected and reconstructed amplitude (Figure 7.7). The mean value of the residual is below 1 ADC-count in the entire high gain range and 2 ADC-counts in low gain. The spread obtained for each injected amplitude is produced by the different phases. If the parabolic correction is applied the largest residuals are produced by pulses with the largest times (Figure 7.5). The absolute value of the residual is proportional to the injected charge as observed in Figure 7.7.

Since the amplitude reconstructed is used to obtain the time of the pulse, the bias produced by phase variations in the amplitude reconstruction affects the estimation of the time to the second order. Figure 7.8a shows the correlation between the phase of the injected pulses and the reconstructed phase. In addition to the bias, the DSP saturates the reconstructed time for

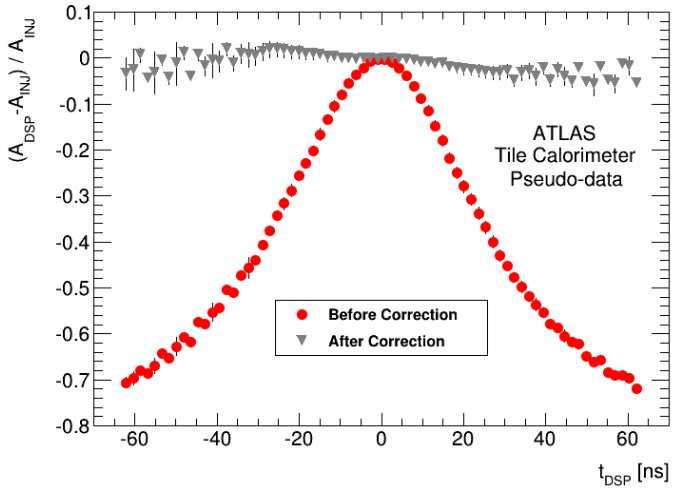


Figure 7.5: Relative difference between injected and reconstructed amplitude as a function of the reconstructed phase before (red circles) and after the parabolic correction (grey triangles).

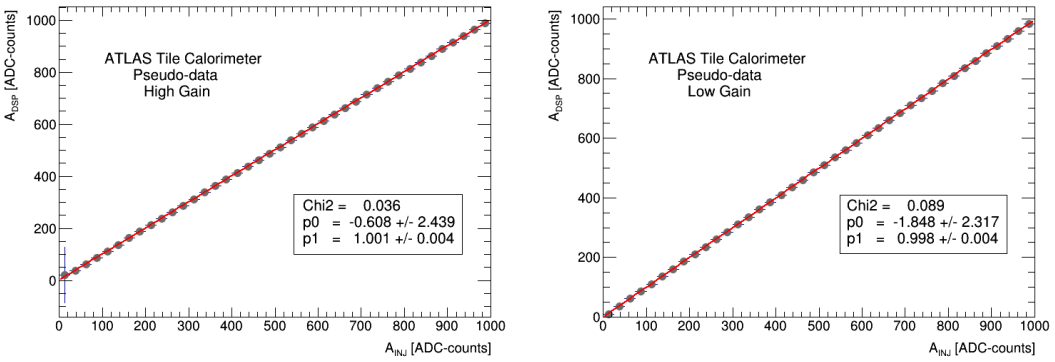
negative energies. For pulses with times within ±12.5 ns (one bunch crossing) the difference between the injected and reconstructed time is constant and close to zero (Figure 7.8b). The residual corresponds to the expected phase of the channel which for this particular channel is 2 ns.

7.3.2 Qualification with Calibration Data

The CIS has been selected to qualify the performance of the DSP reconstruction in the real system. In this case, the injected pulses include the electronic noise introduced by the readout electronics as well as the characteristic leakage pulse of the CIS (Figure 6.1). In addition, the CIS introduces an uncertainty between the injected charge configured and the actual charge produced in the electronics [17].

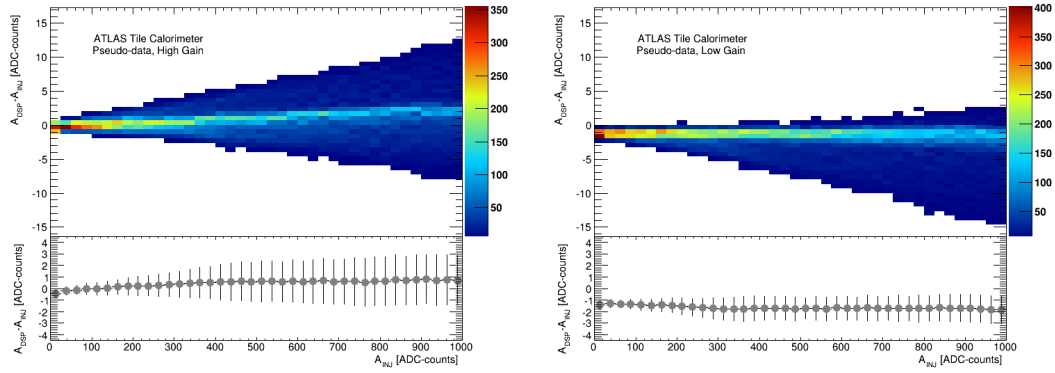
A CIS run with special settings was used to inject pulses with different amplitudes in the two gain ranges and with a fixed phase equal to zero (Table 7.2). The injected charge is given by the equation:

$$Q = 2C\frac{4.096 \times N_{DAC}}{1023} \tag{7.1}$$



(a) (b)

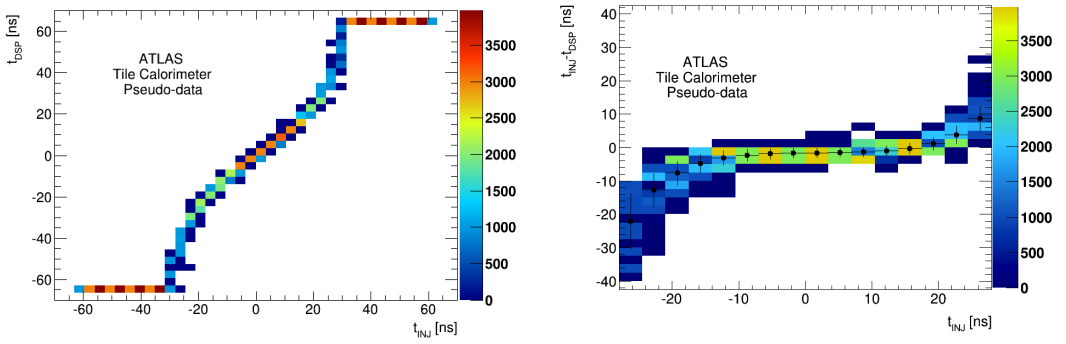
Figure 7.6: Correlation between injected and reconstructed amplitude with parabolic correction for pulses within ±12.5 ns for high (a) and low (b) gains.



(a) (b)

Figure 7.7: Residual of the difference between injected and reconstructed amplitude with parabolic correction for pulses within ±12.5 ns for high (a) and low (b) gains as a function of the injected amplitude. The bottom part of the plots show the average and the RMS.

where C is the capacitor (5 pF or 100 pF) and N_{DAC} is the DAC value which corresponds to the voltage used to charge the capacitor. A 5 pF capacitor is used for the high gain range and a 100 pF for the low gain in this analysis. The high gain covers a range from 0 to 12 pC whereas the low gain goes from 13 to 800 pC.



(a) (b)

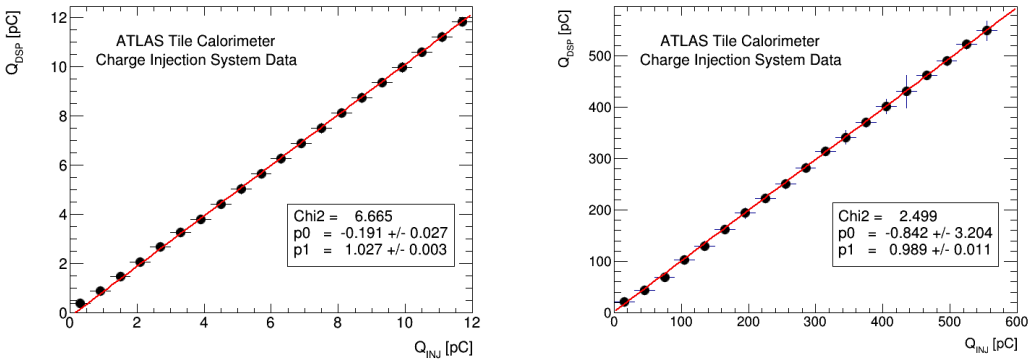
Figure 7.8: Correlation (a) and absolute difference (b) between injected and reconstructed phase for pulses in the entire amplitude range for one channel with an expected phase of 2 ns.

Run number	219572
Amplitude DAC	[0,1020] DAC counts
Amplitude step	1 ADC-count
Number of events per charge	50
Time	0 ns
Capacitor	5 pF, 100 pF
Total events	102000

Table 7.2: Parameters of the CIS run 219572 used for the qualification of the DSP reconstruction.

The analysis is performed using one single channel in order to disentangle the effects of using channels with different expected phases and different Optimal Filtering weights from the effects of the reconstruction. In particular channel 1 of module LBA01 was used in this study.

Figure 7.9 shows the reconstructed charge (Q_{DSP}) versus the injected charge (Q_{INJ}) for high and low gains in pC units. The parabolic correction is not applied since the pulses are injected with a phase equal to zero although small phase variations are produced. The parameters of the linear fit presented in Figure 7.9 show a better agreement between injected and reconstructed charge for low gain due to a larger effect of the constant leakage pulse in the high gain.



(a) (b)

Figure 7.9: Correlation between the injected and reconstructed charge for high (a) and low (b) gains using the CIS.

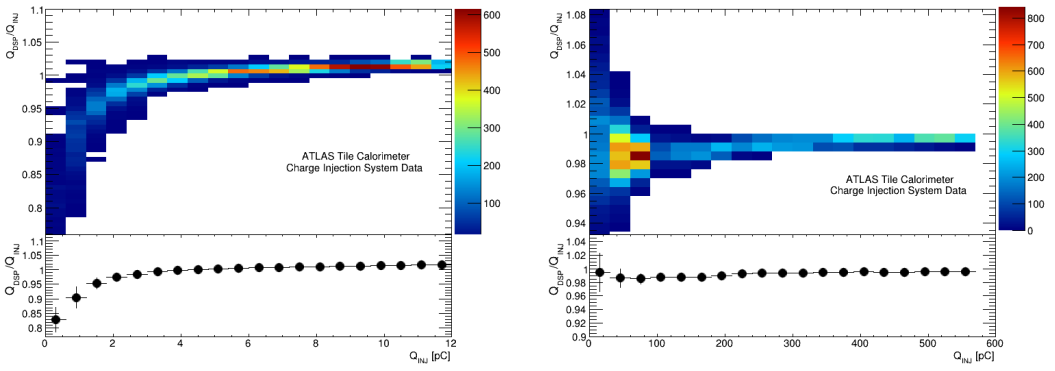
The ratio between the reconstructed charge and the injected charge is presented in Figure 7.10. Again, the effect of the leakage pulse is more pronounced at small injected charges and specially in the high gain range. In both gains this effect is minimal for injected charges in the second half of the gain range. For charges above 7 pC in the high gain and above 100 pC in the low gain the agreement between injected and reconstructed charge is within 1% which corroborates the good agreement between the calibration of the CIS and the reconstruction algorithms.

The run used included 50 pulses for each injected charge in the range. The resolution can be defined as the RMS divided by the mean value of the reconstructed charge for each charge (Figure 7.11). The resolution is better than 1% for charges above 0.5 pC in the entire range for both gains. This result proves that the DSP reconstruction does not deteriorates the TileCal design resolution given by:

$$\frac{\sigma E}{E} = \frac{50\%}{E} \bigoplus 1\% \tag{7.2}$$

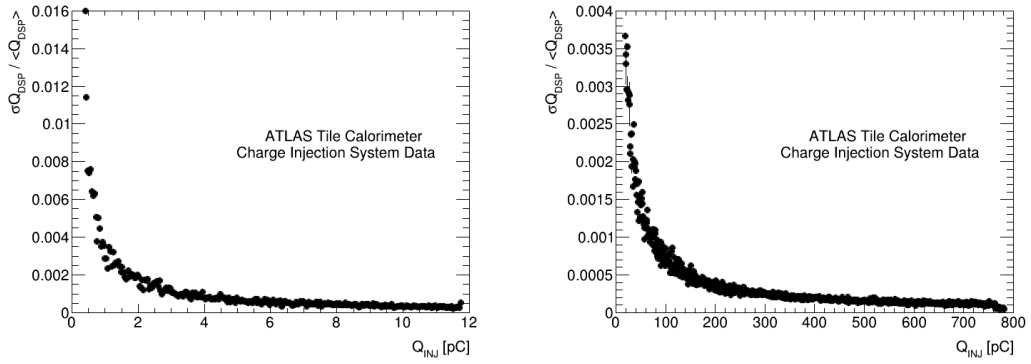
for a normal calibration constant to MeV the obtained resolution implies a variation of 4 MeV at 1 GeV.

The residual can be defined as the relative difference between the injected and recons-



(a) (b)

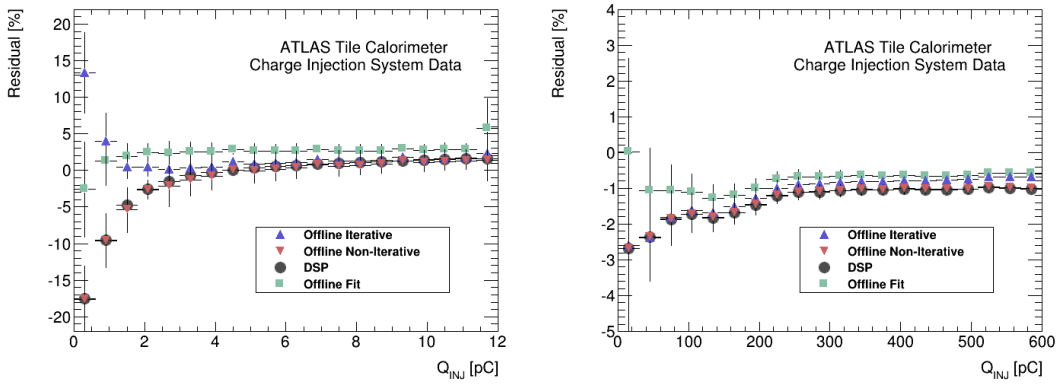
Figure 7.10: Ratio between injected and reconstructed charge as a function of the injected charge for high (a) and low (b) gains using the CIS. The plot in the bottom part shows the average and RMS.



(a) (b)

Figure 7.11: Resolution of the DSP reconstruction as a function of the injected charge for high (a) and low (b) gains using the CIS.

tructed charges. Figure 7.12 shows the residual as a function of the injected charge for high (a) and low (b) gains for the four reconstruction methods available. The Optimal Filtering non-iterative algorithm in the DSP (black circles) and offline (red triangles) show a perfect agreement in the entire range. Moreover, the Optimal Filtering iterative (blue triangles) is also in agreement within 1% with the DSP result for charges above 4 pC in the high gain

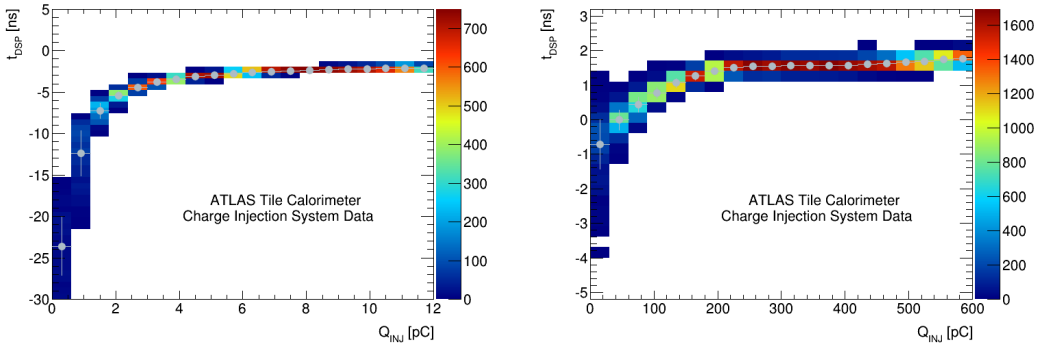


(a) (b)

Figure 7.12: Residual as function of the injected charge for high (a) and low (b) gains using the CIS for the DSP Optimal Filtering (black circles), offline non-iterative Optimal Filtering (red triangles), offline iterative Optimal Filtering (blue triangles) and the Fit (green squares) methods.

and for the entire low gain range. The residual increases for low charges in the high gain range mainly caused by the effect of the leakage pulse. This effect is even more pronounced when comparing with the Fit method (green squares). Nevertheless, the agreement between the methods in better than 1% except for low charges in the two gain ranges where the effect of the constant leakage pulse of the CIS is considerable. This result proves that the effect of using fixed point arithmetic in the DSP is negligible and the reconstruction is compatible with other algorithms running offline with floating point processors.

The CIS was configured to inject pulses with zero absolute time, i.e. the fourth sample in the peak of the pulse. However, the time is adjusted at digitizer level (groups of 6 channels) and therefore small channel to channel variations are expected. Moreover, the time reconstructed by the DSP corresponds to the time between the expected phase in the database for each channel and the actual phase of the pulse. Figure 7.13 shows the reconstructed time as a function of the injected charge. Apart from the low charge region in the two gain ranges, which again are affected by the leakage pulse, the reconstructed time is constant and close to zero. The obtained constant value corresponds to the residual of the channel and can be used



(a) (b)

Figure 7.13: Reconstructed time as a function of the injected charge for high (a) and low (b) gains using the CIS data. The grey points show the average and the RMS.

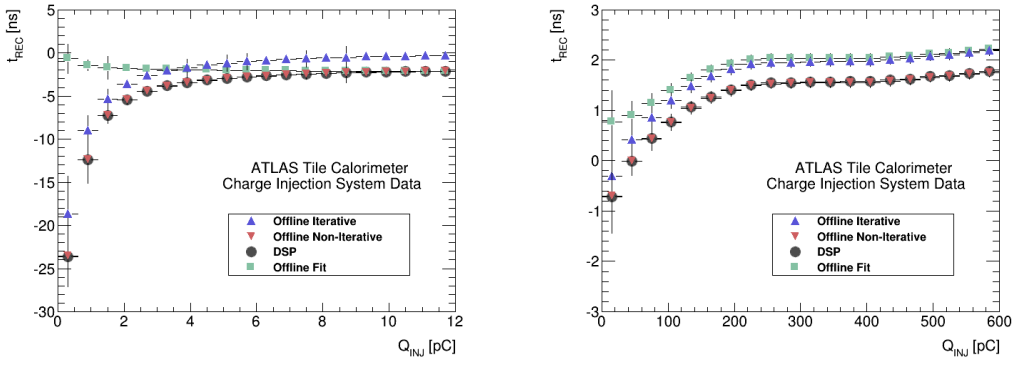
to correct the expected phase in the database for this particular channel in order to obtain a reconstructed time equal to zero.

Moreover, the time reconstructed by the DSP is consistent within 1 ns with the other reconstruction methods for charges above 4 pC in the high gain and in the entire low gain range (Figure 7.14).

7.4 Optimal Filtering Performance with LHC Collisions Data

7.4.1 Timing Adjustments for Collisions

The OF algorithm is based on the assumption that pulses have small, known and with a fixed phase. In the previous Section it was shown how the variations in the phase affects the amplitude reconstruction. The calibration of the timing in the TileCal refers to the hardware and software adjustments to ensure a small, known and fixed phase. First, the phase of the digitizing clock used in the front-end is configured to have the central sample close to the peak for the triggered pulses. The configuration is done for groups of six channels connected to the same Digitizer board.



(a) (b)

Figure 7.14: Reconstructed time as a function of the injected charge for high (a) and low (b) gains using the CIS for the DSP Optimal Filtering (black circles), offline non-iterative Optimal Filtering (red triangles), offline iterative Optimal Filtering (blue triangles) and the Fit (green squares) methods.

The residual phase of each single channel corresponds to the phase of a pulse produced by a particles coming from the interaction point. This expected phase is measured and stored in the conditions database and is used to select the correct set of Optimal Filtering weights for each channel. The method used to measure the expected phase uses physics collisions data. However, not all the pulses produced during physics runs are generated by particles coming from the interaction point or from the triggered bunch crossings (out-of-time pileup). Therefore, the method used selects jets with an associated track in the inner detector in order to guarantee energy depositions produced by particles coming from the interaction point of the triggered bunch crossing. Then, those channels with a deposited energy above 0.5 GeV and connected to cells inside a cone of size R=0.4 around the jet track are selected. Figure 7.15 shows the time distribution for all channels with energy above 500 MeV (blue) and for channels inside the jet cone (dashed line).

Ideally, the average reconstructed time for the selected channels must be zero. The expected phase stored in the conditions database is corrected if variations greater than $\pm$ 3 ns are observed in a channel. Variations below this threshold are not considered because they have a negligible impact in the signal reconstruction since the parabolic correction is applied.

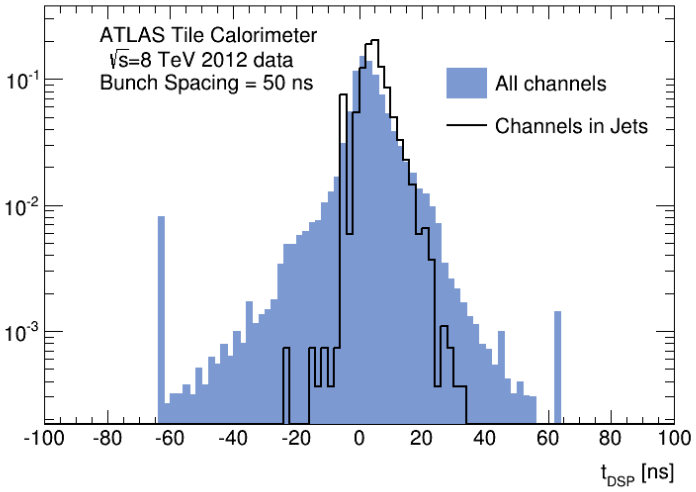


Figure 7.15: Time distributions for all channels with reconstructed energy above 0.5 GeV (blue) and for channels inside a jet (black line) for a 2012 collisions run with 50 ns of bunch spacing.

For the analysis, a map with the average time of each channel is built (Figure 7.16) and the average time of each channel extracted. The phase reconstructed by the DSP corresponds to the time between the peak of the pulse and the expected phase used to computed the weights. Therefore, the average time reconstructed by the DSP is the correction that should be applied to the expected phase stored in the database. A similar histogram is used in the online monitoring to detect global timing shifts at the level of digitizer or module. These global timing shifts are usually caused by a wrong configuration of the module. The timing map is used to detect and reconfigure modules or digitizers with timing problems during data taking.

7.4.2 Performance of the DSP Reconstruction with Low Pileup Collisions Data

During 2010 the LHC operated with a reduced number of colliding bunches and therefore with large empty gaps between bunches. Due to the absence of out-of-time pileup, any

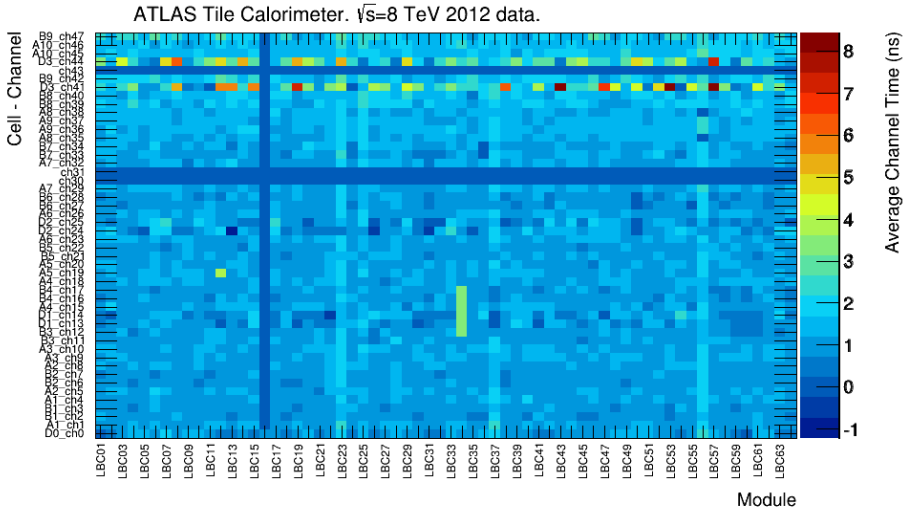


Figure 7.16: Average time per channel for LBC partitions in a 2012 physics run. Vertical line correspond to a powered off module whereas the horizontal lines are non instrumented channels.

deposited signal within the seven samples transmitted to the RODs corresponded to the triggered event. Thus, the parabolic correction to the online reconstructed energy was applied for all the pulses, whereas the default offline reconstruction method was the iterative Optimal Filtering algorithm. Figure 7.17 shows the distribution of energy as a function of time for the offline iterative and the DSP non-iterative methods with the parabolic correction applied for the *JetTauEtmiss* stream of ATLAS run 167776 (Table 7.3). The distributions for the two methods show a similar behavior for this type of collisions. High energy depositions mainly produced by jets present a better defined timing close to zero whereas electronic noise, beam gas and other background particles have a slow wider time distribution with low energy depositions.

The offline iterative method presented the best behavior for this type of isolated depositions and it was used as reference to validate the DSP reconstruction. The bias produced by time variations affects the DSP reconstruction while the iterative method is not affected (Figure 6.13). However, for high energy depositions the time resolution is better (Figure 7.17)

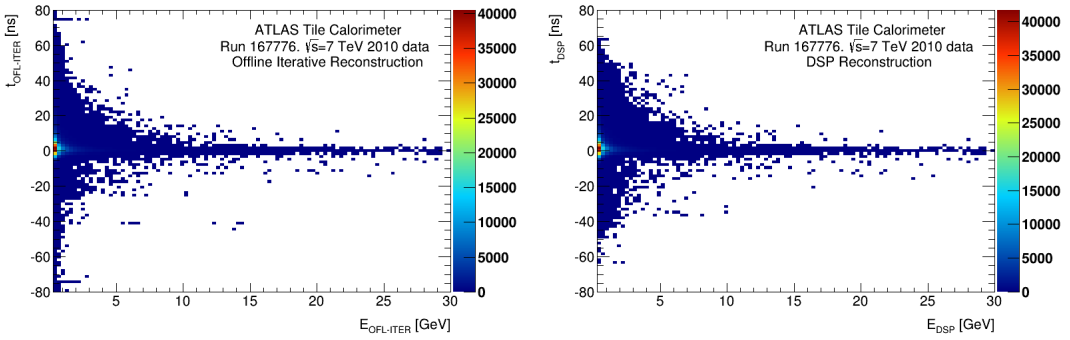


Figure 7.17: Reconstructed energy as a function of time for run 167776 (2010, low pileup). The plot in the left shows the energy and time reconstructed with the offline iterative methods whereas the plot in the right corresponds to DSP reconstruction magnitudes.

Run number	167776
Integrated luminosity	6293 nb^{-1}
Colliding bunches	348
Peak luminosity	1.8 x $10^{32} cm^{-1} s^{-1}$
Bunch spacing	150 ns
Peak $< \mu >$	3.31

Table 7.3: LHC parameters for run 167776 of 2010 used to validate the DSP performance with low pileup.

and the two methods are compatible if the parabolic correction is applied to the DSP reconstruction.

Figure 7.18 shows the ratio between the energy reconstructed with the offline iterative method and with the DSP non-iterative with the parabolic correction applied. The agreement between the two methods is very good in average although there are some discrepancies at low energy due to scattered time distribution.

The Optimal Filtering method reconstructs $A\tau$ which must be divided by the amplitude to obtain the phase. Therefore, the bias produced by time variations in the amplitude affects directly to the phase result. Essentially, the amplitude is underestimated due to phase variations and this effect overestimates the reconstructed phase.

Figure 7.19 shows the correlation between the reconstructed time offline with the iterative

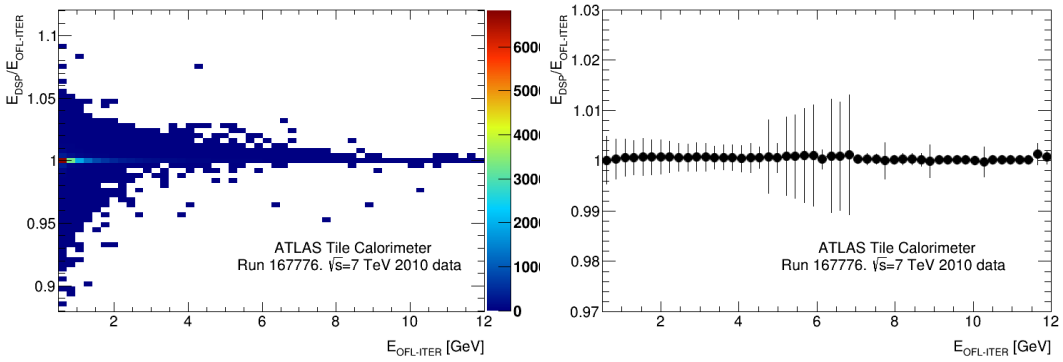


Figure 7.18: Ratio between the energy reconstructed with the offline iterative algorithm and the DSP non-iterative method as a function of offline energy for collisions run 167776 (2010, low pileup). The plot in the right shows the average and RMS of the ratio.

method, not affected by phase variations, and the reconstructed time by the DSP. The two methods show a good agreement for pulses with small phases while for signals above 30 ns the time reconstructed by the DSP increases rapidly. Nevertheless, the design bunch spacing of the LHC is 25 ns and depositions of energy for larger times can be considered as not produced by the triggered event.

The differences observed between the offline iterative and the DSP non-iterative are dominated by the bias produced by phase variations in the DSP method. A minimization of the bias selecting in-time pulses or a different offline method has to be used in order to characterize the precision of the DSP fixed point arithmetic with physics data. Figure 7.20 shows how the DSP result is consistent with the offline Optimal Filtering iterative algorithm if pulses with a reconstructed phase within ± 1 ns are selected. In this case, the deviation produced by phase variations is minimized and the two algorithms are compatible.

On the other hand, Figure 7.21 shows the difference between the reconstructed energy by the DSP and offline using the same non-iterative Optimal Filtering algorithm. The observed difference comes from the different precision in the arithmetic (fixed in the DSP versus floating point offline) and from the different precision of the calibration constants. The DSP packs the calibration constant to convert ADC to MeV units in 16-bit words whereas 32-bit values are used in the offline case. This Figure also shows the expected limited precision for a typical

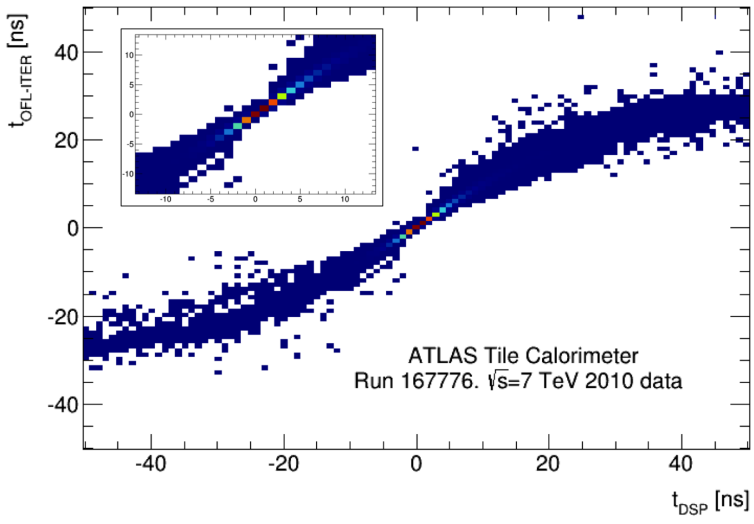


Figure 7.19: Correlation between the time reconstructed offline with the iterative method and in the DSP with the non-iterative method. The small plot in the corner shows the time range [-12.5, 12.5] ns.

calibration constant, containing 99% of the TileCal channels (red dashed line), and for the worst case of the highest calibration constant in the detector (blue dashed line).

The same consideration can be used to study the reconstruction of the time in the DSP. However, the treatment of negative amplitudes in the reconstruction of the time is different in the DSP and offline, even if the same Optimal Filtering non-iterative algorithm is used. In the DSP, the division by the amplitude to obtain the time is implemented using a Look Up Table (LUT). Since negative amplitudes are caused by pulses above |40| ns (Figure 7.4a), the time of pulses with negative reconstructed amplitude is saturated to the maximum value. The time of pulses with negative reconstructed amplitude and positive $A\tau$ result is forced to +64 ns whereas if $A\tau$ is negative the reconstructed time will be -64 ns. In the offline case, the $A\tau$ term is always divided by the amplitude to obtain the time even if the reconstructed amplitude is negative. In order to compare the two reconstruction methods the DSP approximation is emulated in the offline result for negative amplitudes.

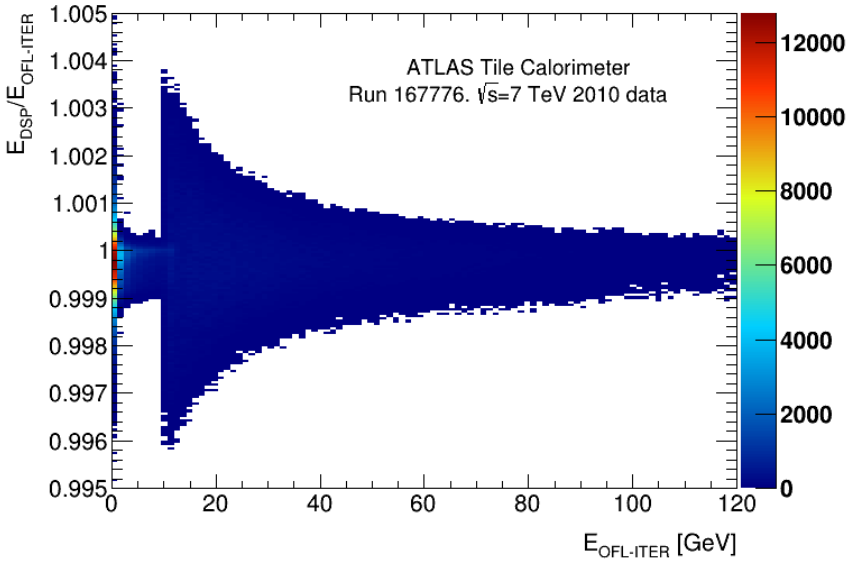


Figure 7.20: Ratio between the reconstructed energy in the DSP and the offline iterative methods as a function of the offline energy for pulses with absolute time smaller than 1 ns.

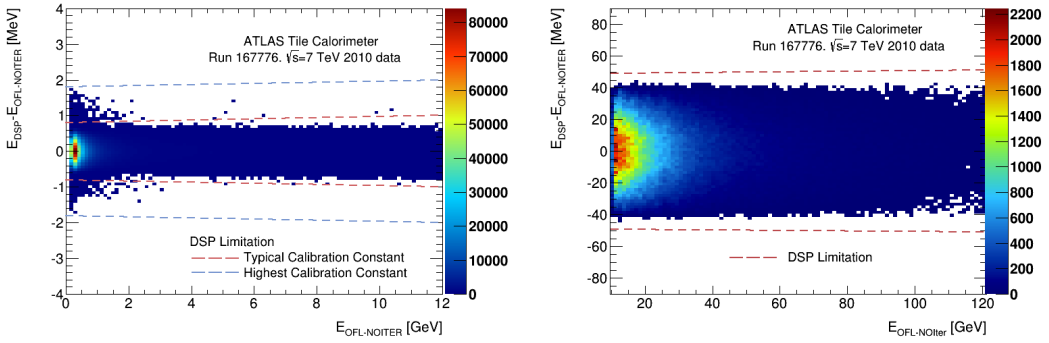


Figure 7.21: Absolute difference between the energy reconstructed in the DSP and offline with the non-iterative method as a function of the energy reconstructed offline for high (left) and low (right) gains. The dashed lines show the expected limited precision for typical calibration constants.

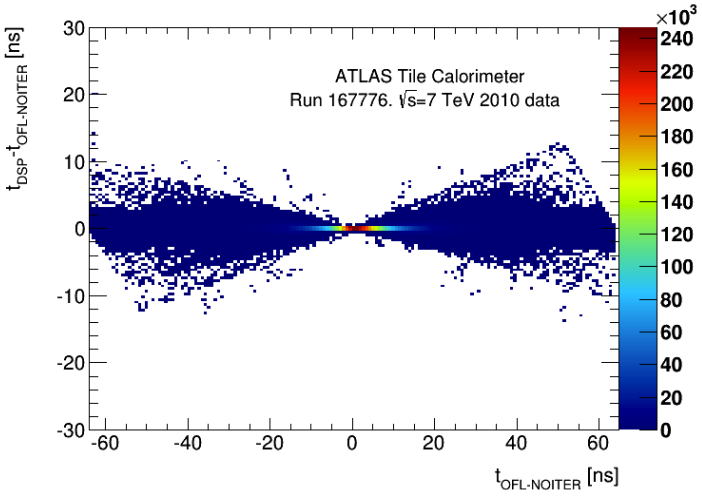


Figure 7.22: Absolute difference between the time reconstructed by the DSP and offline with the non-iterative method as a funtion of the offline time. The effect of the LUT is enhanced for pulses with large times.

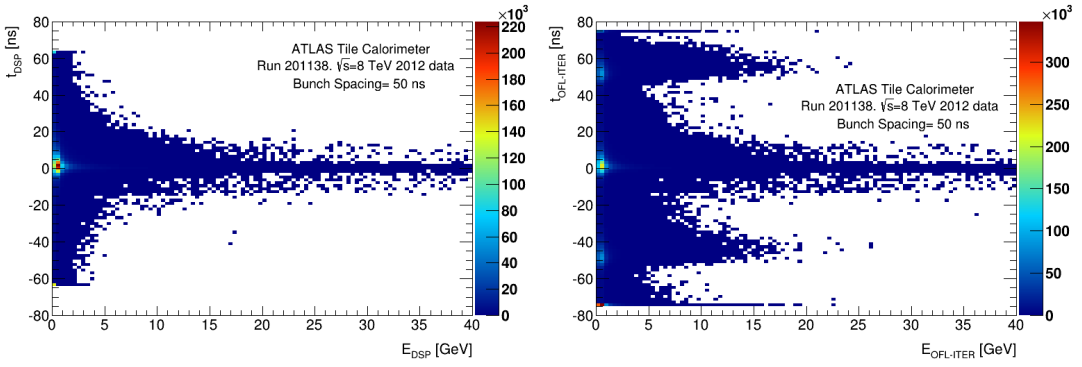
Apart from the negative amplitudes treatment, the main difference between the two methods comes from the different implementation of the division by the amplitude, using a LUT in the DSP and a floating point division in the offline. This effect is enhanced for large phases as shown in Figure 7.22. For pulses with phases within ±1 bunch crossing (±25 ns) the maximum difference obtained between the two methods is ±3 ns. Furthermore, more than 95% of the pulses in this collisions run have a time in the [-6,+6] ns range and the maximum difference between the two methods is ±1 ns in this range which is compatible with the jitter of the LHC clock.

7.5 Optimal Filtering Reconstruction with 50 ns Bunch Spacing Collisions

In order to increase the peak luminosity the number of bunches in the LHC ring was increased in 2011 and the minimum bunch spacing was reduced to 50 ns. Due to that, the TileCal signal

reconstruction strategy was changed and the Optimal Filtering non-iterative method became the default offline method replacing the iterative algorithm used during commissioning and first year of LHC operation. This decision was taken to minimize the impact of pileup.

Figure 7.23 shows the reconstructed energy as a function of the reconstructed time using (a) the DSP and (b) the offline iterative algorithms. The iterative method reconstructs the pulse with maximum amplitude within the 7 samples window (±75 ns) which causes the peaks at ±50 ns in Figure 7.23b. The non-iterative method only considers depositions around the central sample. Pulses above ±40 ns are reconstructed as negative energies and can be considered noise.



(a) (b)

Figure 7.23: Distribution of the energy as a function of the time reconstructed with (a) the DSP and (b) the offline iterative methods for physics run 201138 with a minimum bunch spacing of 50 ns.

This effect is also observed in the distribution of the reconstructed time for the offline iterative and non-iterative and the DSP methods (Figure 7.24). Again, in order to compare the offline and DSP non-iterative methods the treatment of negative amplitudes in the DSP is emulated in the offline case. The non-iterative method considers the pulses around the central sample (zero reconstructed time) whereas the iterative algorithm also reconstructs the pulses produced in different bunch crossings (peaks at ±50 ns).

With the new reconstructed strategy adopted by TileCal, since the same non-iterative method is used, the only difference between the DSP and the offline reconstruction comes

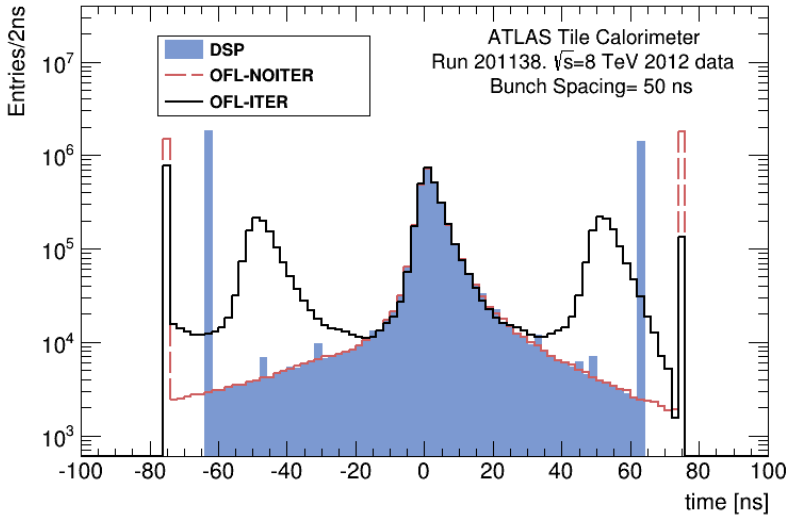
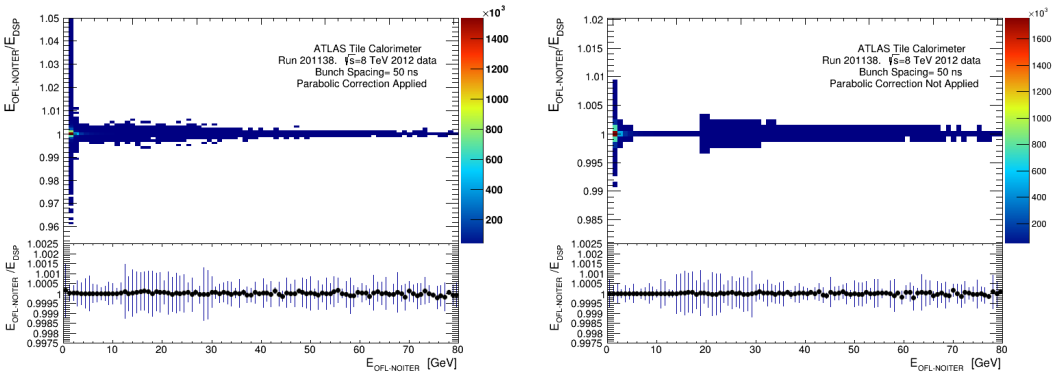


Figure 7.24: Distribution of the reconstructed time using DSP, offline iterative and non-iterative methods for the physics run 2011138 with a minimum bunch spacing of 50 ns.

from the different arithmetic and precision of the calibration constants. Figure 7.25 shows an agreement better than 1% between the amplitude reconstructed by the DSP and offline for both gains. The parabolic correction can still be applied for small phase variations but in a reduced range of phases not to correct the depositions of energy produced in a different bunch crossing. By default, the parabolic correction is applied only for pulses within half of the minimum bunch spacing. During 2011 and 2012 the minimum bunch spacing in the LHC was 50 ns and therefore the parabolic correction was applied only to pulses within ± 25 ns.

The parabolic correction can be applied to the energy reconstructed in the DSP and offline. In both cases the correction is applied offline using floating point arithmetic. The difference in the reconstructed time by the two methods produces a different result for the correction. Thus, the difference between the energy reconstructed by the DSP and offline after applying the parabolic correction increases (Figure 7.25a). The difference between the methods after the correction increases more for larger times because the difference in the reconstructed time is also larger as shown in Figure 7.22. Moreover, the difference after the correction is larger



(a) (b)

Figure 7.25: Ratio between the energy reconstructed with the DSP and offline non-iterative methods as a function of the offline energy with (a) and without (b) parabolic correction applied.

in the low energy region because the time distribution is wider. Nevertheless, the agreement between the DSP and offline methods after the parabolic correction is better than 1% in the entire energy range for energies above 0.5 GeV. Figure 7.25 also shows the mean and the RMS of the ratio between the DSP and offline results. In this case the agreement is below 0.1% in the entire range. This result summarizes the difference between the DSP reconstruction result, mainly used in the HLT, and the offline reconstruction used for physics analysis.

7.6 Performance of the Quality Factor Reconstruction

The QF can be used to quantify the goodness of the reconstruction, for instance to detect pileup noise or data corruption. It is essentially an estimation of the expected samples for the reconstructed amplitude and time assuming a known and constant pulse shape and the difference with respect to the actual samples. In ideal conditions the expected and real samples are equal and the QF is zero. However, variations in the pulse shape (electronics noise, pileup, large phase variations, etc...) produce an increase of the difference and in the QF result. The expression to compute the quality factor was presented in the previous Chapter (Equation (6.14)). The implementation in the fixed point arithmetic of the DSP includes two

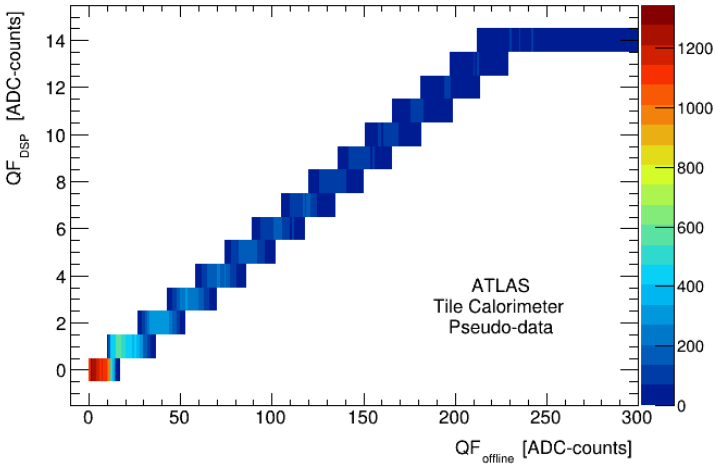
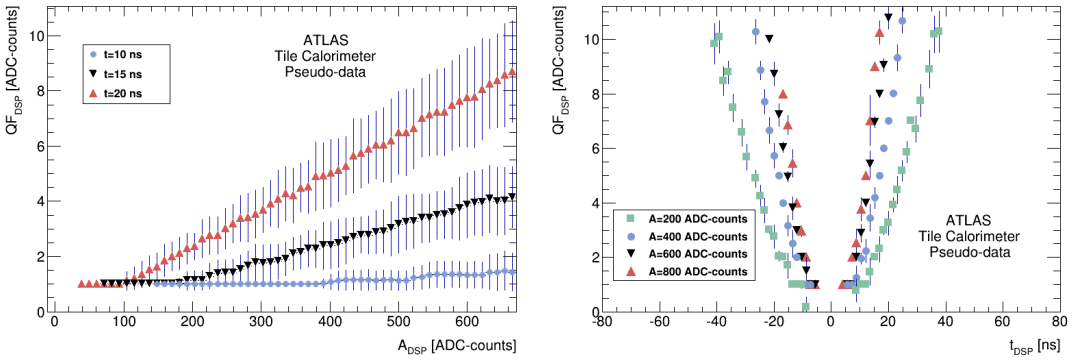


Figure 7.26: Correlation between the Quality Factor computed in the DSP and offline for pseudo-data.

main differences with respect to the offline case; first, the square root is implemented using a LUT; second, the DSP result is packed using only 4-bits (range from 0 to 14 where value 15 is reserved to tag other types of problems) and a reduction scale is used to accommodate the range. Since the offline quality factor varies in the range from 0 to 255, it was decided to use a scale to be sensitive in the same range but with reduced precision. Basically, the result of the quality factor of the DSP is divided by $\sqrt{512} = 22.63$ with respect to the offline. The reduction scale and the limited precision of the DSP with respect to the offline quality factor is shown in Figure 7.26.

There are two features of this definition of the QF that are also observed in the DSP result. First, the QF increases with the amplitude since it is proportional to the absolute difference between expected and actual samples. Then, even if the derivative of the pulse is used to correct the variation in the pulse time, the QF increases with large time variations.

Figure 7.27a shows the dependency with the amplitude for different injected times and Figure 7.27b shows the time dependency for different amplitudes. Note that these results were obtained with ideal pulses which does not contain neither electronic nor minimum bias noise.



(a) (b)

Figure 7.27: a) Result of the QF as a function of the reconstructed amplitude in the DSP for different injected times. b) Result of the QF as a function of the reconstructed time in the DSP for different injected amplitudes. In both cases it corresponds to pseudo-data.

The dependency with the energy both for offline and DSP methods using real collisions data with 50 ns of minimum bunch spacing is presented in Figure 7.28. The electronics and pileup noise produce a wider distribution although the dependency is still clear for both methods. The slope is more abrupt in the case of negative energies with a faster saturation of the QF.

A study performed using simulated data with various levels of pileup showed that the QF can be used to discriminate between signal with and without out-of-time pileup (Figure 7.29) [48]. The offline QF was used in the study and it shows that for signals up to 12 GeV the variation of the quality factor in the presence of out-of-time pileup goes from around 10 to 35 ADC-counts.

The conclusion is that the reduction scale used in the DSP to adjust the offline to the limited DSP range needs a revision in order to have sensitivity to the variations showed in this study. Since the square root is implemented in the DSP using a LUT, the scale does not need to be linear and other functions might be considered in order to have various ranges with different precision.

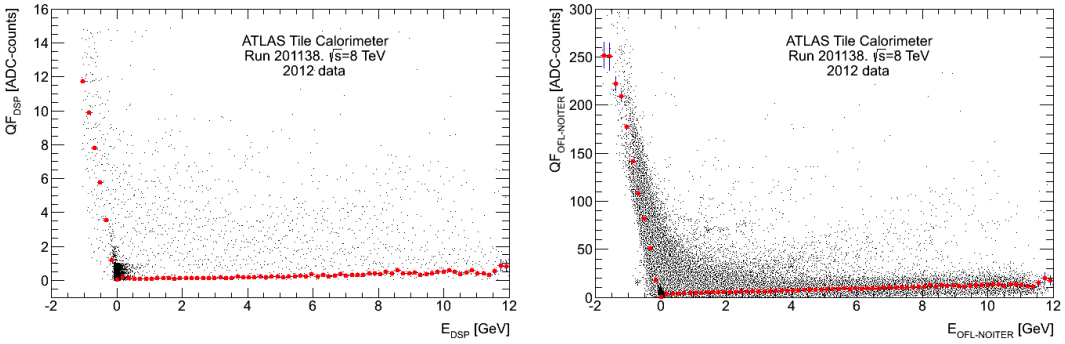


Figure 7.28: QF as a function of the reconstructed amplitude for DSP (left) and offline (right) for the physics run 201138.

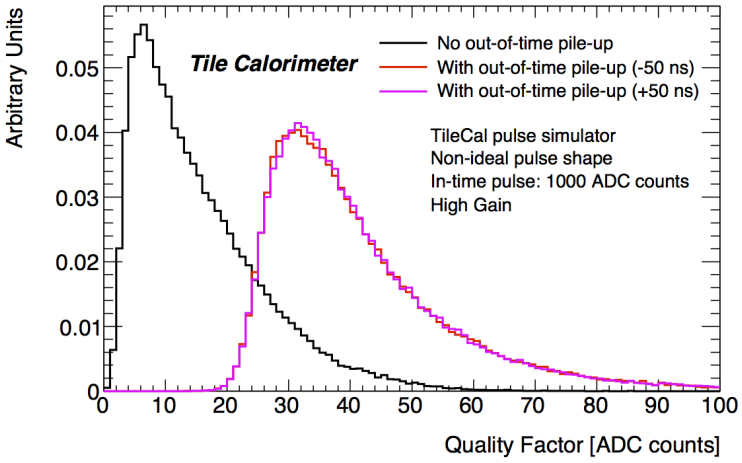


Figure 7.29: Normalized distributions of offline QF for simulated data. Generated amplitude of in-time pulse is 12 GeV. Amplitude of the out-of-time pulse follows the distribution in ZeroBias stream with a cut on 34 ADC-counts. Three cases: No out-of-time pileup (black), with out-of-time pileup (-50 ns), with out-of-time pileup (+50 ns).

7.7 Study of the DSP Reconstruction Using Jets

7.7.1 ATLAS Jet Reconstruction Algorithms

The official jet reconstruction method in ATLAS is the so-called $anti-k_t$ algorithm [49]. The reconstruction of a jet is performed starting from a list of more basic constituents and the

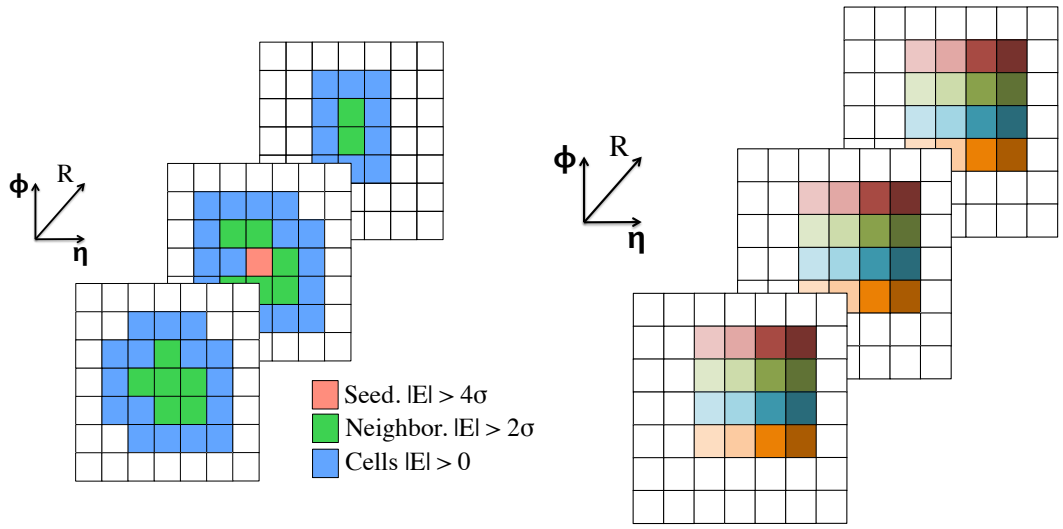


Figure 7.30: Sketch of cluster (left) and tower (right) structure in three consecutive layers of the calorimeter within a jet cone of R=0.4. The cells are considered inside the cluster depending on the energy:noise ratio. The towers include all the cells with the same eta coordinate, represented with the same color in the sketch).

momentum of the jet before applying any calibration can be defined as:

$$P_{jet} = \sum P_{constituents} \tag{7.3}$$

In this algorithm, the constituents or inputs to the jet algorithm can be either calorimeter towers or 3D topological clusters. The calorimeter towers include all the cells within a 0.1×0.1 square in (η,ϕ) and therefore the constituents of the jet (towers) have a fixed size (Figure 7.30). On the other hand, the topological clusters are formed following the *4/2/0* algorithm. The method starts with the selection of a seed cells with $|E|>4\sigma$, where σ corresponds to the characteristic noise of the cell (electronic + pileup). The cluster is also formed by the seed neighbour cell with $|E|>2\sigma$ and finally by all neighbour cells with $|E|>0$ (Figure 7.30). Therefore, the topological structure of the constituent of a jet reconstructed using clusters is not fixed and depends on the cell energy depositions and cell noise. In addition to the constituents, it is possible to define the radius size (R) of the jet cone around the selected track. A size of R=0.4 has been used in this analysis both for towers and clusters.

In this Section, the impact of the channel energy estimation using the DSP algorithm in the final jet reconstruction will be studied. The two methods of jet reconstruction based on towers and clusters will be analyzed, and the DSP and offline methods results will be compared . Jet variables like multiplicity, size and total energy are studied as a function of the TileCal channel reconstruction method.

Jet selection and methodology The aim is to study and quantify the impact of using the DSP or the offline methods to reconstruct the TileCal signals in the reconstruction of jets in ATLAS. The offline methods used are the iterative and non-iterative Optimal Filtering algorithms. The iterative method represents a better reconstruction method in absence of pileup since it is not biased by phase variations. On the other hand, the non-iterative algorithm is used as a reference to quantify the effect of the fixed point arithmetic of the DSP. In order to compare the DSP and offline methods under the same conditions, the jet reconstruction algorithms were used starting from the same raw data sample of a particular low pileup run (Table 7.3) and changing only the channel reconstruction method in TileCal. This allows the comparison of the different jet reconstructed magnitudes in a event by event basis. Moreover this low pileup collisions run (bunch spacing larger than 100 ns) has been selected in order to diminish other effects like pileup or high jet multiplicity.

Only the most energetic jet with an associated track in the inner detector and with at least one cell in TileCal is considered in each event. It is also required that the jet is present for both reconstruction algorithms in a R<0.2 area and must also be isolated (no additional jets in a R<0.8 area around the track).

7.7.2 Impact of the DSP Energy Reconstruction in Jets

For each event the difference in the number of reconstructed jets between online and offline reconstruction methods in TileCal is used to study the impact of the DSP method at a first order. As expected, the jet multiplicity is more consistent between the DSP and the offline non-iterative method than with respect to the iterative one (Figure 7.31).

The distribution of the energy for all jets in the data samples using the DSP and offline reconstruction methods for TileCal signals are shown in Figure 7.32. After the jet selection the

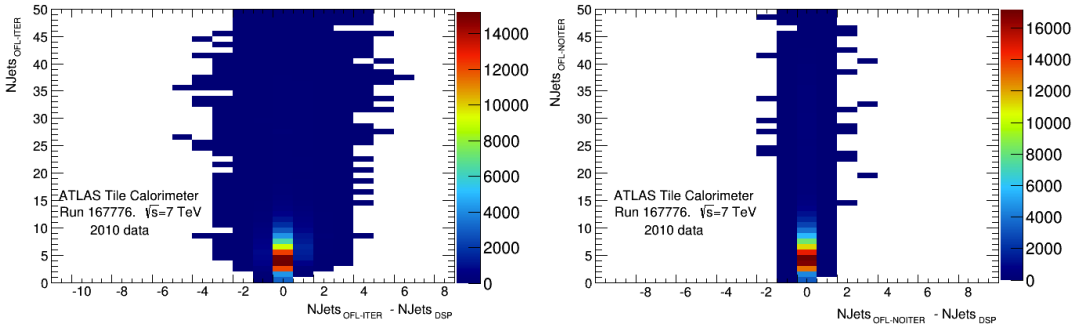


Figure 7.31: Difference between the number of jets per event using the DSP and the offline iterative (left) and non-iterative methods (right).

DSP and the offline non-iterative and iterative methods are overall in good agreement. The result is similar using towers and clusters as input parameter in the jet *anti-kt* reconstruction algorithm. The energy of the jets is mainly concentrated to the region below 200 GeV even if it exceeds 1 TeV for each jet.

The main effect of using towers instead of clusters is shown in Figure 7.33, which shows the number of TileCal cells within a jet using the DSP method as a function of the offline iterative one. The fixed structure of the towers show a better correlation between both methods with respect to the clusters. Since the inclusion of a cell in a cluster depends on the ratio between the energy and the noise of the cell, a small variation in the reconstructed energy may lead to the inclusion or exclusion of a cell in a cluster.

The difference at the channel level between the DSP and offline iterative methods was previously studied in this Chapter, showing a considerable difference due to the bias produced by phase variations in the DSP results, even if the parabolic correction is applied. This effect was more pronounced in the low energy region where the time of the depositions was shown to have a wider distribution. Furthermore, this low energy region is critical in the cluster formation since the noise thresholds are in this range of energies. The effect is reduced if the DSP is compared with the offline non-iterative method, but it is still considerable for clusters with respect to towers (Figure 7.34).

The different number of cells inside a jet shows a difference in the total energy of the jet.

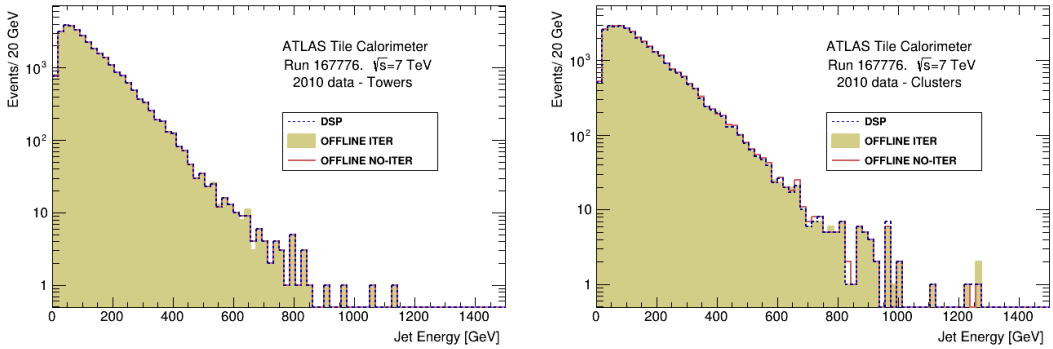


Figure 7.32: Jet energy distribution using the DSP and offline iterative and non-iterative method to reconstruct TileCal signals using towers (left) and clusters (right) as input to the jet reconstruction algorithms for low pileup data.

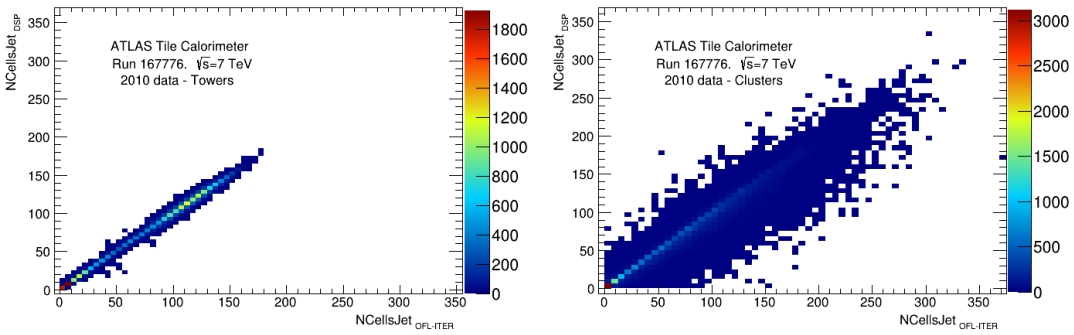


Figure 7.33: Correlation of number of cells inside a jet using the DSP and the offline iterative methods if towers (left) and clusters (right) are used as input parameter to the jet reconstruction algorithms.

Figure 7.35 shows the relative difference in the total energy of the jet between the DSP and offline iterative methods for TileCal signals.

The difference due to the bias produced by phase variations in the DSP is enlarged by the clustering effect with respect to the jets built from towers. However, the plots include at the bottom part the mean of the difference and the RMS, showing that in average, the methods are compatible within 1% even if clusters are used as input to the jet reconstruction algorithm.

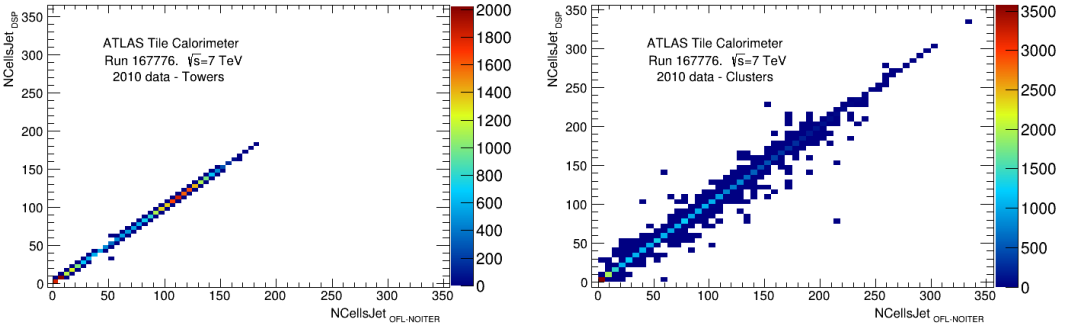


Figure 7.34: Correlation in the number of cells inside a jet using the DSP and the offline non-iterative methods if towers (left) and clusters (right) are used as input parameter to the jet reconstruction algorithms.

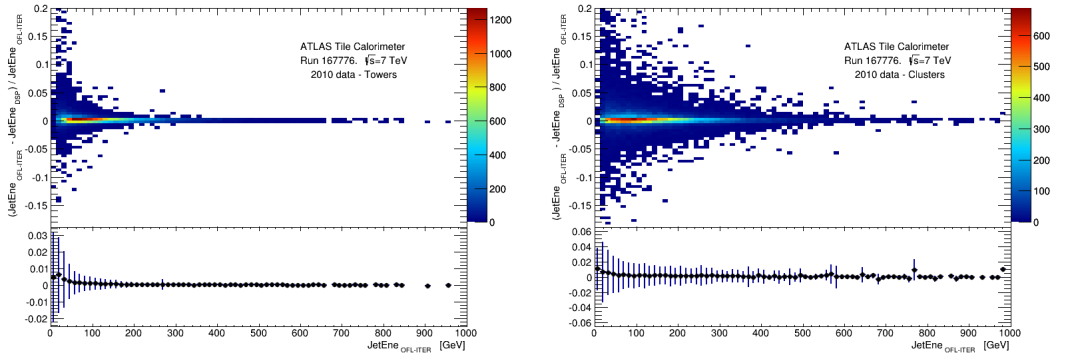


Figure 7.35: Relative difference in the total jet energy between the DSP and offline iterative methods as a function of the jet energy reconstructed with the offline iterative method using towers (left) and clusters (right) as input parameter for the jet reconstruction for low pileup data. The bottom part shows the average and RMS.

The DSP and the offline non-iterative methods show an even better agreement with few outliers with a maximum difference of 5%, but with average differences below 1 per mil in the whole range (Figure 7.36).

The effect of the clustering is more evident in the correlation between the absolute difference in the total jet energy and the fraction of this difference produced in TileCal cells (Figure 7.37). Since only the TileCal signal reconstruction method is changed, we would expect that the main change in the total energy of a jet is produced in the TileCal cells.

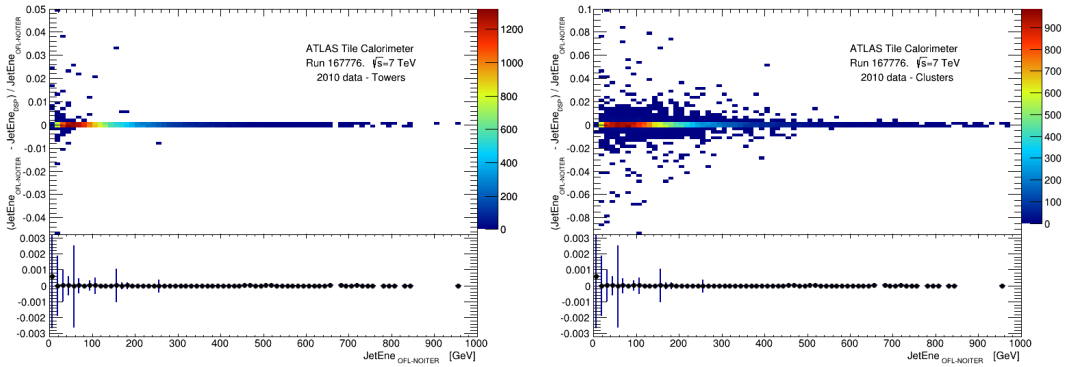


Figure 7.36: Relative difference in the total jet energy between the DSP and offline non-iterative methods as a function of the jet energy reconstructed with the offline non-iterative method using towers (left) and clusters (right) as input parameter for the jet reconstruction. The bottom part shows the average and RMS.

This assumption is valid in the case of towers as input parameters for the jet reconstruction where the difference in the total jet energy is mainly produced by TileCal cells. In the case of clusters used as jet constituents, the difference in the energy of Tilecal cells can produce a change in the structure of the cluster which can be propagated into the LAr calorimeter. As observed in Figure 7.37, small variation in the TileCal cells leads to large variations in the total jet energy.

Impact of the DSP Reconstruction with High Pileup Data

As mentioned earlier in this Chapter, with the reduction of the bunch spacing, the Optimal Filtering non-iterative became the default offline reconstruction in TileCal in order to minimize the effect of the out-of-time pileup. Thus, in order to study the effect of the DSP reconstruction, the offline Optimal Filtering non-iterative method must be used as reference. Run 201138 was selected to perform this study (Table 7.4).

The increase in the peak luminosity and the number of interactions per crossing causes an increase in the multiplicity of reconstructed jets which minimizes the number of isolated jets as defined previously. The jets selected for the analysis with high pileup must not include another jet in the same event in a cone of R<0.4 (instead of R<0.8 requested in the low

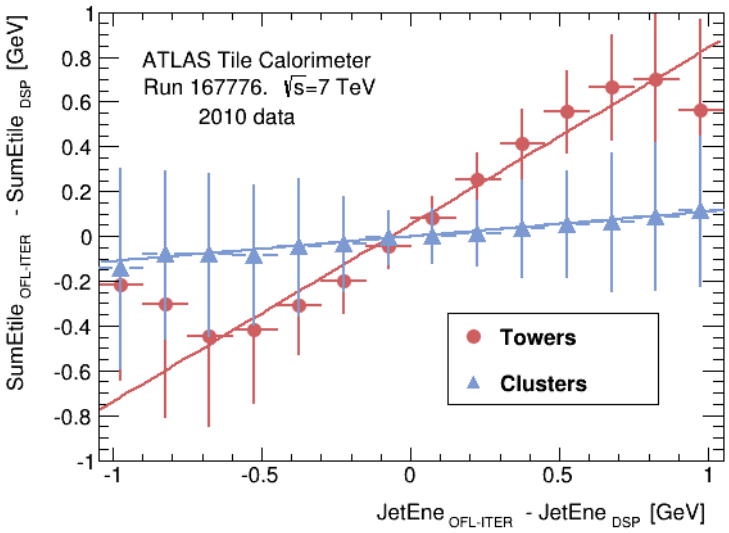


Figure 7.37: Absolute difference in the total jet energy between DSP and offline iterative method versus the difference produced in the Tile calorimeter cells using towers (blue) and clusters (red) as input parameters to the jet reconstruction algorithm.

pileup case). This new requirement is used to increase the statistics and allows the study of additional effects in the final jet reconstruction. Therefore, the comparison between the DSP and offline methods will reflect not only the differences produced by the reconstruction at the TileCal channel level (already studied in the previous Section) but also the side effects produced by the pileup in the final jet reconstruction.

Run number	201138
Integrated luminosity	4.57 x 10^4 nb^{-1}
Colliding bunches	618
Peak luminosity	2.2 x 10^{33} $cm^{-1}s^{-1}$
Bunch spacing	50 ns
Peak $< \mu >$	22.5

Table 7.4: LHC parameters for run 201138 of 2012 used to validate the DSP performance with high pileup.

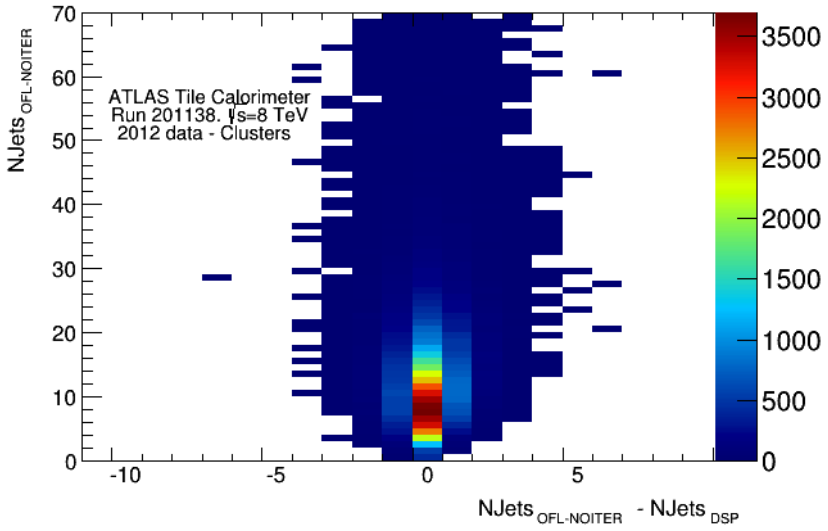


Figure 7.38: Difference in the number of reconstructed jets per event between the DSP and the offline iterative methods as a function of the number of jets reconstructed with the offline method for high pileup data.

The absolute difference between the number of online and offline reconstructed jets per event slightly increases with respect to the low pileup case (Figure 7.38). However, the difference between methods is negligible since both in the DSP and offline the same non-iterative method are used.

Figure 7.39 shows the distribution of the jet energy using the DSP and offline non-iterative methods for towers and clusters.

The distributions show a considerable increase in the energy of the jets compared to 2010 data (Figure 7.32). Nevertheless, the DSP and offline non-iterative methods are in overall in good agreement, specially in the case of towers as jet constituents, as was already shown for low pileup data.

The effect of the increase of pileup is more evident by comparing the total jet energy for the DSP and offline methods. Figure 7.40 shows the relative difference of the total jet energy using the DSP and offline non-iterative methods as a function of the offline jet energy for

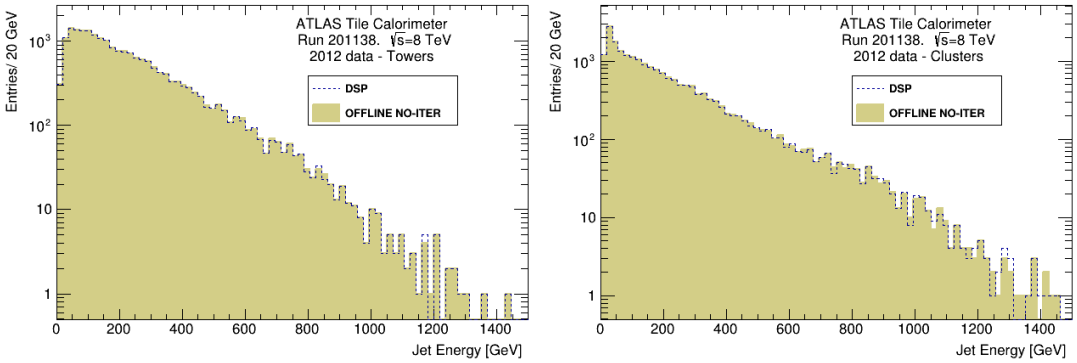


Figure 7.39: Distribution of the jet energy using the DSP and the offline non-iterative methods for towers (left) and clusters (right) as input parameters for the jet reconstruction for high pileup data.

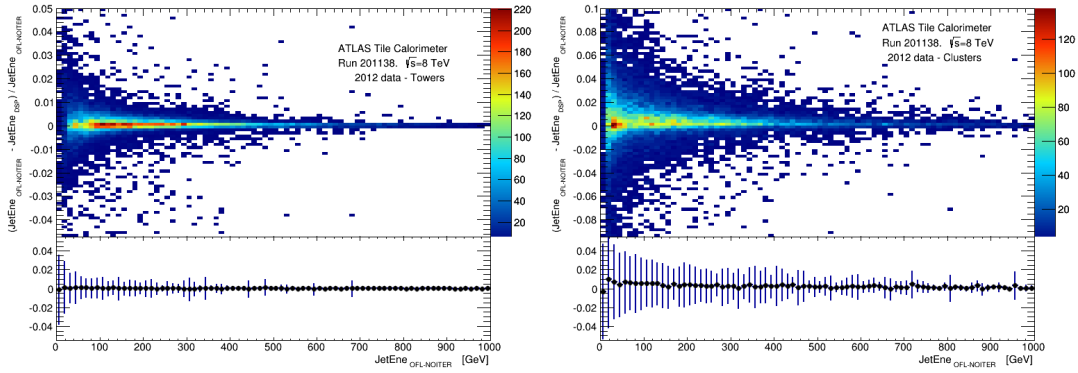


Figure 7.40: Relative difference in the total jet energy between DSP and offline iterative methods as a function of the jet energy reconstructed with the offline iterative method using towers (left) and clusters (right) as input parameter for the jet reconstruction for 2012 high pileup data. The bottom part shows the average and RMS.

towers and clusters. The difference is larger than in the low pileup case (Figure 7.36) and also is larger if clusters are used as jet constituents with respect to towers. Nevertheless, the average difference between both methods (lower part of Figure 7.40) shows an agreement between them close to zero with an RMS smaller than 5% for both clusters and towers.

Chapter 8

Conclusions

The first part of this Thesis is devoted to introduce the readout electronics of TileCal. It includes a detailed description of the TileCal back-end system and specially the ROD and OMB modules. Then, the production, installation and performance results during the first three years of operation of the back-end system are presented. The second part is focused in the description of the TileCal signal reconstruction algorithms used in the back-end system, in particular the Optimal Filtering method. It includes a detailed description of the Optimal Filtering algorithm and some performance results using different type of data.

A total of 32 ROD modules are needed to readout the whole TileCal sub-detector, and in case of a considerable increase in the SEU errors, 32 OMB boards would be needed to improve the data quality efficiency. Therefore, a total of 38 RODs and OMBs (32 plus 6 spares) were produced to operate the detector. A dedicated test-bench was designed to reproduce the detector operation conditions and each ROD and OMB board was qualified after passing a set of tests. The results presented in Chapter 5 show a BER better than 10^{-10} at a 95% of confidence level for the back-end electronics system.

The installation of the ROD modules started in the summer of 2005 and the complete system was ready in September 2008. The ROD modules were used to certify the correct installation of the front-end electronics by processing calibration and muon cosmic data. The first LHC collisions arrived at the ATLAS experiment in November 2009 with a center of mass

energy of 900 GeV. In 2010 the LHC started to produce luminosity for the experiments at a 7 TeV center of mass energy. Due to the low instantaneous luminosity of the LHC, the TileCal RODs operated in a special mode were both reconstruction and the complete raw data were transmitted, which allowed the validation of the DSP reconstruction methods. The increase of the instantaneous luminosity in the following two years forced a change in the readout strategy and only part of the digital samples were transmitted out of the RODs to reduce the data packet size. During these three first years of operation, the data taking efficiency of ATLAS was kept above 93%. The contribution of the TileCal back-end system to the total ATLAS DAQ inefficiency was compatible with the rest of the sub-systems, representing around 10% of the total deadtime. A total of 28.9 fb^{-1} were delivered by the LHC with a total of 26.9 fb^{-1} recorded by the ATLAS experiment.

The Optimal Filtering algorithm is the main signal reconstruction method used in TileCal. It provides the amplitude and time of the digitized samples received in the RODs from each PMT. In addition, the RODs calibrate the reconstructed amplitude to energy, provide an estimator of the goodness of the reconstruction (quality factor), compute the total energy sum and its transverse and z projection for each front-end module which is used in the Level 2 trigger system and pack part of the digital samples. Chapter 6 describes these reconstruction algorithms and their implementation in the DSPs of the ROD modules.

The validation and qualification of the reconstruction in the DSPs are studied in Chapter 7 using different type of data. First, the implementation of the method in the DSPs is validated using pseudo-data generated in the OMB with ideal pulse shapes. The results confirmed the correct implementation of the method and only the expected differences between the DSP and offline methods due to the different point arithmetic used were observed.

Then, the CIS was used to study the performance of the DSP reconstruction with data generated in the front-end electronics. Despite the effect of the characteristic leakage pulse of the CIS, the results show an agreement better than 1% between the injected and reconstructed charge. Moreover, it shows a resolution of the reconstruction better than 1% for charges above 0.5 pC, which proves that the DSP reconstruction does not deteriorate the TileCal design resolution. The reconstruction of the time is also studied with compatible results between the DSP and three different offline reconstruction methods.

Finally, the impact of using the DSP with respect to the offline reconstruction results is studied using collision data, first at the channel level and also with more complex physics objects like jets. The results show that if the same reconstruction algorithm is used in the DSP and offline, which was the default operating method for most of the data acquired in the first three years of operation, the differences between the two methods are below 1% in the entire energy range. Concerning the time reconstruction, the differences are larger mainly because of the usage of a LUT to implement the division in the fixed point arithmetic of the DSP. Nevertheless, the differences between the DSP and offline are below 3 ns in 98% of the pulses. The impact in the reconstruction of jets in ATLAS depends on the constituents used to build jets. If pseudo-rapidity towers, with predefined size and topology are used to construct the jets, the differences between using the DSP and the offline reconstruction in TileCal are smaller compared to the usage of clusters, with variable size and topology. Nevertheless, the differences in the total energy of jets between using the DSP and offline methods are below 1% in average with an RMS smaller than 5%.

Capítulo 9

Resumen

9.1 El Experimento ATLAS en el Gran Colisionador de Hadrones en el CERN

El CERN es el Consejo Europeo para Investigación Nuclear (acrónimo francés de *Conseil European pour la Recherche Nucleaire*). El nombre proviene del consejo provisional creado para establecer el laboratorio en 1952 con la misión de instaurar una organización europea para la investigación de física fundamental.

Hoy en día, el entendimiento de la matería nos lleva mucho más adentro del núcleo, por lo que la principal área de investigación en el CERN es ahora la física de partículas, el estudio de los constituyentes fundamentales de la materia y las fuerzas que actúan sobre ellos. El descubrimiento de los bosones W, Z y recientemente de Higgs se pueden considerar los resultados más notables del CERN desde su fundación. Pero además, el CERN ofrece un entorno inmejorable para el desarrollo de nuevas tecnologías en campos como la electrónica o la computación.

El Gran Colisionador Electron-Positron (LEP, *Large Electron-Positron collider*), en funcionamiento entre los años 1989 y 2000, consistía en un acelerador circular con una circunferencia de 27 km construido en un túnel cercano al CERN cruzando las fronteras de Suiza y Francia. Ese mismo túnel fue utilizado posteriormente para construir el Gran Colisionador de Hadrónes

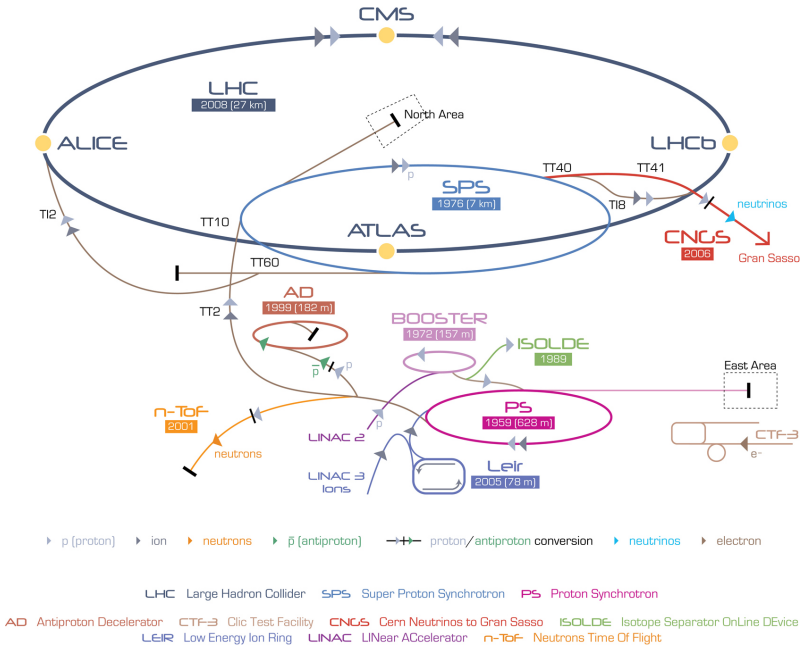


Figura 9.1: Esquema del complejo de aceleradores del CERN.

(LHC, *Large Hadron Collider*).

Los protones son acelerados y agrupados en haces en 4 aceleradores previos a su inyección con una energía de 450 GeV en el anillo de 27 km del LHC, como se muestra en la Figura 9.1. Entonces, los haces son acelerados en el anillo a energías de hasta 7 TeV. Cuando la energía final es alcanzada, los haces de protones se hacen colisionar en el centro de cuatro grandes experimentos a una energía en centro de masas de hasta 14 TeV. No obstante, en los primeros años de funcionamiento los haces de protones fueron colisionados a una energía en centro de masas máxima de 8 TeV.

Los cuatros experimentos se encuentran instalados a lo largo del anillo del LHC. ATLAS y CMS son dos experimentos de propósito general ubicados en puntos opuestos en el anillo, con el objetivo principal de estudiar la existencia de nueva física en la escala del TeV. Los otros dos experimentos has sido diseñados para el estudio de física más específica; LHCb investiga la violación de la simetría CP y ALICE estudia el plasma de quarks-gluones mediante la colisión de Pb-Pb y Pb-protón.

Las partículas que interaccionan fuertemente con el campo de Higgs son pensadas mientras que las ligeras interaccionan débilmente. El campo de Higgs tiene al menos una partícula asociada, el bosón de Higgs. Trás tres años de operación, en 2012 los experimentos ATLAS y CMS anunciaron el descubrimiento de una nueva partícula en la región de masa alrededor de 125 GeV, consistente con el boson de Higgs. Los próximos años de operación del LHC deben proporcionar suficiente estadística para determinar la propiedades de esta nueva partícula.

9.2 El Sistema de electrónica de *Back-End* del Calorímetro TileCal de ATLAS

ATLAS (*A Toroidal LHC Apparatus*) es un detector de propósito general diseñado para aprovechar todo el potencial de descubrimiento de nueva física del LHC. Mide aproximadamente 45 metros de largo y algo más de 25 metros de alto y tiene un peso cercano a las 7000 toneladas. La Figura 9.2 muestra un esquema del detector ATLAS. El detector interno, construido alrededor del tubo por el que circula el haz, ha sido diseñado para reconstruir los vértices y trazas de las partículas generadas en las colisiones. Los calorímetros electromagnético y hadrónico deben medir la energía de las distintas partículas. La última capa está formada por el espectrómetro de muones, que mide las trayectorias de las partículas cargadas que atraviesan completamente los calorímetros, y los toroides magnéticos que curvan la trayectoria de las particulas mediante un campo magnético de 0.5 Tesla.

El calorímetro hadrónico de ATLAS ha sido diseñado principalmente para identificar *jets* y medir su energía y dirección así como para medir la deficiencia de energía total transversa (E_T^{miss}). Con el fin de ser sensitivo a la física de interés, una resolución energética de $50\,\%/\sqrt{E}\oplus 3\,\%$ y una segmentación de $\Delta\eta$ x $\Delta\phi = 0.1$ x 0.1 es requerida en la región central. Para la región más delantera, se precisa una resolución $100\,\%/\sqrt{E}\oplus 10\,\%$ con una segmentación de $\Delta\eta$ x $\Delta\phi = 0.2$ x 0.2.

El calorímetro hadrónico de tejas (TileCal) es un detector de muestreo que utiliza acero como material absorbente y centelleador como medio activo. Se situa en la región $|\eta|<1.7$ trás el calorímetro electromagnético de argón líquido, y está subdividido en un barril central de 5.8 metros de longitud, y dos barriles extendidos de 2.6 metros cada uno. Cada barril

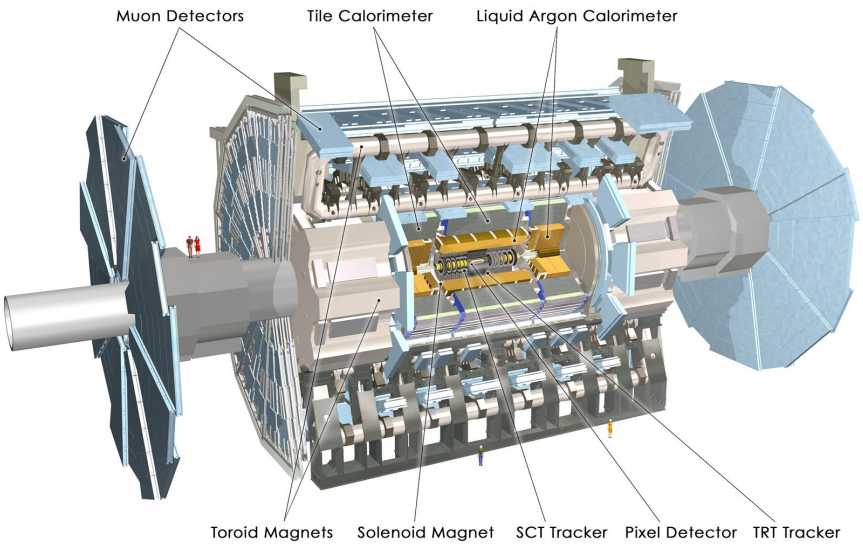


Figura 9.2: El experimento ATLAS.

cilíndrico tiene un radio interno de 2.28 metros y uno externo de 4.25 metros como se muestra en la Figura 9.3.

Las múltiples partículas producidas en la interacciones protón-protón en el centro de ATLAS atraviesan el calorímetro TileCal produciendo luz en las celdas centelleadoras al depositar su energía. La longitud de onda de la luz es transformada al azul visible en la fibras desplazadoras de longitud de onda y guiada a los tubos fotomultiplicadores (PMT) (Figura 9.4). Los fotomultiplicadores generan un pulso analógico con amplitud proporcional a la luz generada en el centelleador y a su vez a la energía depositada por las partículas. Este pulso es amplificado y moldeado en las tarjetas *3-in-1* (Figura 9.5) y por un lado, sumado en grupos de cinco celdas conformando una torre en la dirección η, que son transmitidas al Nivel 1 de *Trigger* de calorímetros (L1Calo). Por otro lado, esos pulsos son recibidos en las tarjetas digitalizadoras donde la señal se muestrea cada 25 ns, con un reloj sincronizado con el cruce de haces del LHC. Las muestras son almacenadas en las memorias tipo *pipeline* en las llamadas Unidades de Gestion de Datos (DMU, *Data Management Unit*). El CTP (Procesador Central de Trigger) se encarga de procesar la información de *Trigger* y selecciona los sucesos que

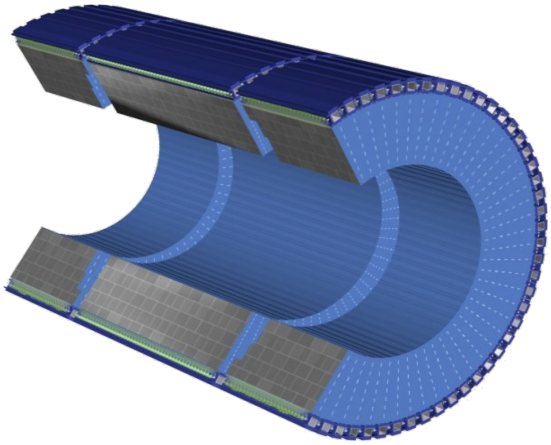


Figura 9.3: Esquema de los tres barriles de TileCal.

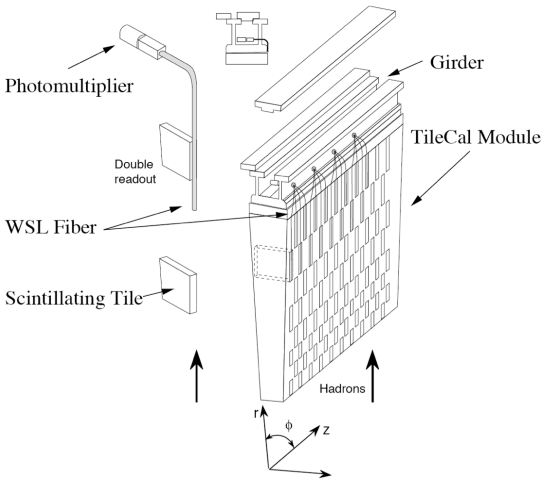


Figura 9.4: Diagrama de un módulo de TileCal y sus componentes.

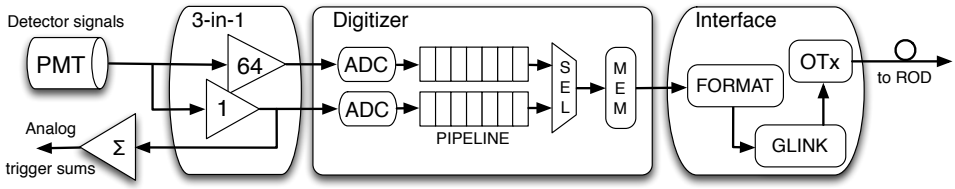


Figura 9.5: Esquema de la electrónica del detector de TileCal.

deben procesarse, con una frecuencia máxima de 100 kHz. Una señal de aceptación de evento (L1A) es enviada a la electrónica del detector a modo de señal de TTC (*Trigger, Timing and Control*). Los datos asociados a estos sucesos son transmitidos a la electrónica de *back-end* del detector para continuar con su procesado.

Las tarjetas *Read-Out Drivers* (ROD) son módulos VME que se encargan de recibir los datos de la electrónica del detector a través de 8 conexiones de fibra óptica, cada una de ellas correspondiente a un módulo del detector (Figura 9.6). Cada módulo ROD está equipado con dos Unidades de Procesado (PU) las cuales se encargan de la reconstrucción de señal en los Procesadores Digitales de Señal (DSP). Acoplado a la parte trasera del ROD se encuentra

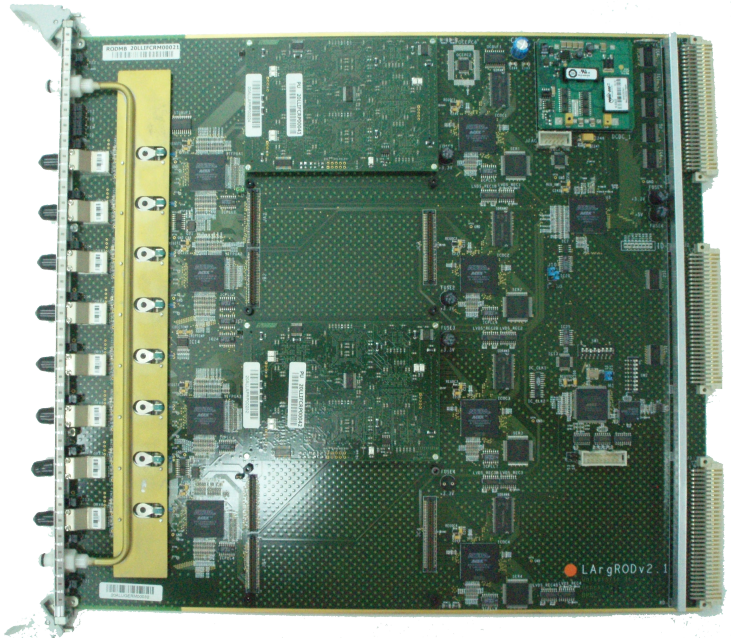


Figura 9.6: Módulo ROD equipado con dos Unidades de Procesado.

un Módulo de Transmisión (TM) que se encarga de transmitir los datos de salida a través de conexiones de fibra óptica.

Test de radiación realizados a la electrónica del *front-end* revelaron una tasa de errores digitales no despreciables al trabajar en un ambiente de radiación como el presente en ATLAS. Con el fin de prevenir la pérdida de datos provocados por esta radiación, se decidió transmitir los datos del detector de manera redundante. Por lo tanto, los datos generados en cada módulo del detector son transmitidos por duplicado a traves de dos conexiones de fibra óptica. La tarjeta *Optical Multiplexer Board 9U* (OMB) ha sido diseñada para recibir los datos redundantes del detector, detectar errores digitales en tiempo real mediante un algoritmo de Comprobación de Redundancia Cíclica y transmitir al ROD los paquetes libres de error. Además, los OMB pueden utilizarse para emular las señales del detector inyectando datos a los RODs para la realización de tests. Las tarjetas OMB han sido diseñadas y producidas por el grupo TileCal del Instituto de Física Corpuscular (IFIC) de Valencia.

9.2.1 Producción y Tests de Validación de los Módulos ROD y OMB

Cada una de los cuatro barriles (dos centrales y dos extendidos) del detector TileCal está asociado a lo que se conoce como una partición en el sistema de adquisición de datos, que pueden operar de manera independiente. Cada partición está formada por un chasis de VME para los módulos ROD y otro para los módulos de TTC. Puesto que cada barril del detector está subdividido en 64 módulos, y cada ROD procesa la información de 8 módulos, cada chasis contiene 8 RODs. Por lo tanto, un total de 32 RODs son necesarios para realizar la lectura de datos de todo el detector TileCal así como 32 OMBs en caso de ser necesarios.

La producción de estos módulos consistió en la fabricación, ensamblaje y validación de 40 unidades de RODs, OMBs y TMs, considerando los 32 necesarios, más 8 unidades de repuesto para cada tarjeta. El banco de pruebas de producción fue diseñado para simular las condiciones de operación de la electrónica de *back-end* de TileCal. Se requería la simulación de las señales provenientes del detector así como las señales de TTC para el control del flujo de datos. Para ello, se equiparon dos chasis de VME, un chasis NIM para la simulación del CTP y un emulador de *Read-Out System* (ROS) para el almacenamiento y análisis de los datos procesados (Figura 9.7).

En el proceso de validación cada módulo ROD debía pasar con éxito cuatro niveles de test. El nivel 0 consistía en un test estático donde se verificaba el acceso por VME a todos los registros de la tarjeta, así como el correcto flujo de datos entre los dispositivos programables (FPGA, *Field Programmable Gate Array*) de entrada y salida. Una vez superado el test estático, se procedía a realizar los niveles 1,2 y 3 de manera secuencial. El nivel 1 consistía en un test de procesado de datos a baja frecuencia, en el nivel 2 se incrementaba la frecuencia de inyección, y el nivel 3 consistía en un *burn-in* test a mayor frecuencia y de al menos 72 horas de duración. Tras la validación de los 40 módulos ROD se obtuvo una tasa de errores de bit (BER, *Bit Error Rate*) mejor de 10^{-10} para un nivel de confianza del 95 % como muestra la Figura 9.8.

La producción de los módulos OMB se realizó con posterioridad a los ROD, adaptando el banco de pruebas pero manteniendo la filosofía de validación. Cada módulo OMB se consideró óptimo tras superar un nivel estático de test similar a los ROD, y 3 niveles dinámicos donde se incrementaba la frecuencia y la duración de los test.

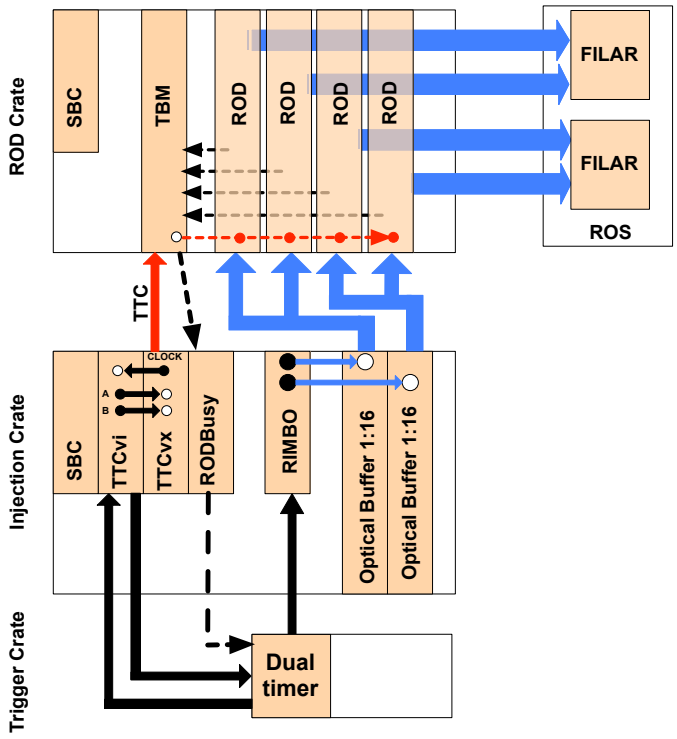


Figura 9.7: Esquema del banco de producción y validación de los módulos ROD.

9.2.2 Instalación y Puesta a Punto

La instalación de la electrónica *back-end* de TileCal en la sala de electrónica de ATLAS (USA15) comenzó en el verano de 2005, simultáneamente con la instalación de los módulos y electrónica del detector (Figura 9.9).

En primer lugar se realizo una réplica del banco de pruebas utilizado para la validación de los módulos ROD, con el fín de emular las señales del detector mientras este era instalado e instrumentado. A medida que los módulos del detector eran instalados y estaban disponibles para su puesta a punto, se fueron conectando a los RODs para su certificación mediante el procesamiento de datos generados por los distintos sistemas de calibración, así como con la toma de datos de muones cósmicos.

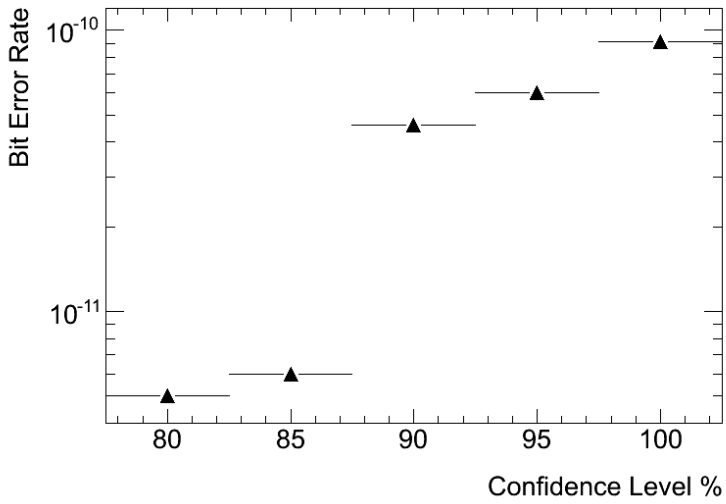


Figura 9.8: Tasa de errores de bit (BER) en función del nivel de confianza para el sistema ROD obtenido durante los tests de validación de 40 módulos ROD.

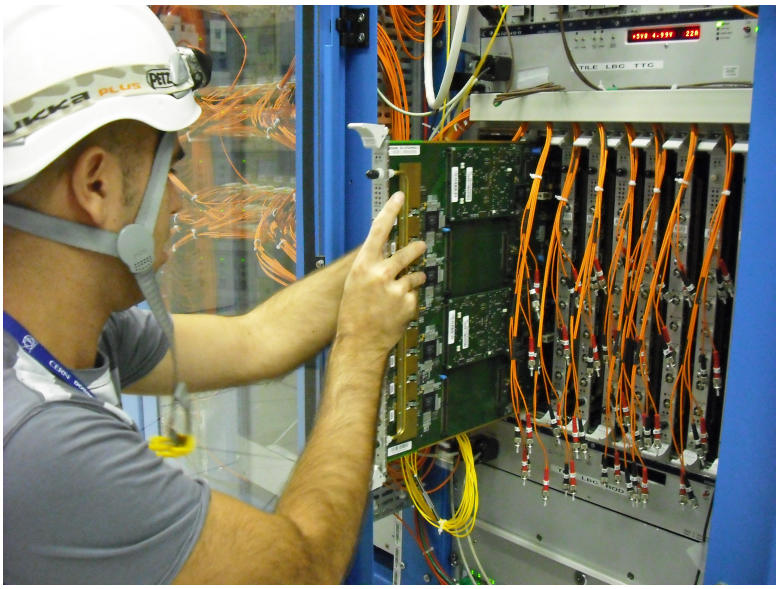


Figura 9.9: Instalación de los módulos ROD en USA15.

Conforme se completaba la instalación del resto de subdetectores de ATLAS, se realizó la integración de todos ellos en el sistema global de adquisición de datos. Está integración se llevó a cabo mediante la adquisición de datos cósmicos (a baja frecuencia), a los que se añadían eventos aleatorios con el fin de aumentar la frecuencia de *Trigger* de primer nivel y emular así las condiciones esperadas con colisiones. El sistema de *back-end* de TileCal funcionó estable durante todos los tests y con un tiempo muerto despreciable hasta una frecuencia de *Trigger* de primer nivel de 45 kHz (la máxima esperada para el primer año de toma de datos) y transmitiendo tanto los datos reconstruidos como las muestras digitales recibidas del detector. Asímismo, en los test de alta frecuencia se alcanzaron frecuencias cercanas a los 100 kHz, en este caso transmitiendo los datos reconstruidos y una parte reducida de las muestras digitales. Por lo tanto, el sistema de *back-end* de TileCal se encontraba a mediados de 2008 preparado para la toma de datos de colisiones y únicamente pequeñas modificaciones de firmware fueron introducidas durante los primeros años de toma de datos.

9.2.3 Funcionamiento Durante los Tres Primeros Años de Toma de Datos de ATLAS

En Septiembre de 2008 comenzó a funcionar el LHC, aunque pocos días después de su puesta en marcha un incidente en uno de los imanes del sector 3-4 obligó una parada de más de un año. Finalmente, en 2010 el LHC comenzó a producir luminosidad para los experimentos a una energía en centro de masas de 7 TeV. La luminosidad instantánea fue incrementada a lo largo del año desde un valor inicial de 0.0016 x 10^{30} $cm^{-2}s^{-1}$ (con un solo haz de protones de baja intensidad en el acelerador) hasta los 2.1 x 10^{32} $cm^{-2}s^{-1}$ a final de ese año con 348 haces en el acelerador y un espacio mínimo entre haces de 150 ns. Estas condiciones de baja luminosidad instantánea a principios de año producían frecuencias de *Trigger* de primer nivel también inferiores a los de diseño en ATLAS, incluso utilizando umbrales de energía para selección de objetos relativamente bajos. Esto permitió operar los RODs en un modo de funcionamiento transmitiendo tanto los datos reconstruidos como las muestras digitales. Las muestras digitales eran reconstruidas de nuevo en el software de reconstrucción *offline* de ATLAS y utilizadas como referencia para evaluar y certificar la reconstrucción realizada en los DSPs de los RODs. La frecuencia de *Trigger* de primer nivel se fue incrementando a medida que la luminosidad

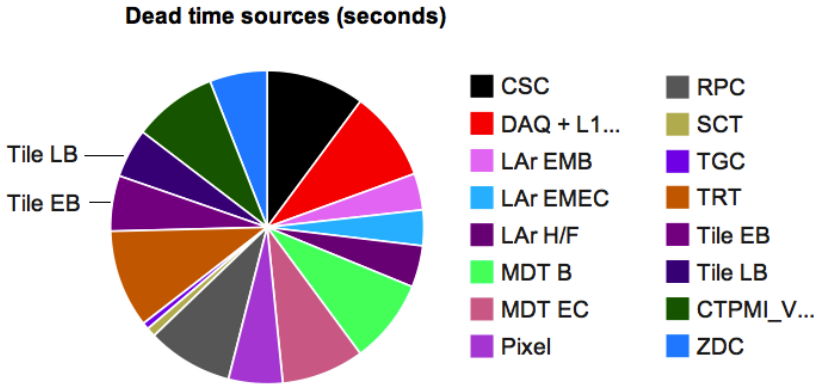
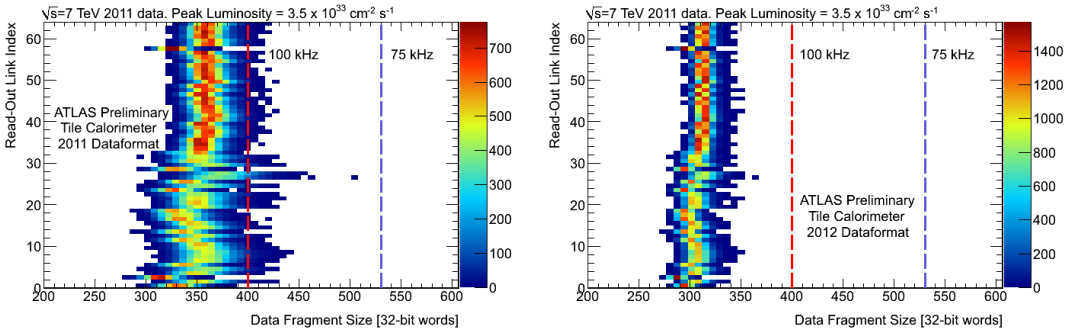


Figura 9.10: Porcentaje de tiempo muerto por subdetector en ATLAS para el año 2010.

instantánea lo hacía, y se precisaron actualizaciones en el fimrware de los ROD con el fin de reducir el tiempo muerto producido. Con todo, el LHC entregó una luminosidad integrada en 2010 de 48.1 pb^{-1} de los cuales 45.0 pb^{-1} fueron guardados por ATLAS, con una eficiencia de adquisición del 93.5 %. TileCal contribuyó un 10 % a la ineficiencia de adquisición total de ATLAS, un valor similar al resto de subdetectores como muestra la Figura 9.10.

En 2011 la luminosidad instantánea continuo aumentando con la inclusión de un mayor número de haces en el acelerador y el aumento del número de protones en cada uno de ellos. Se llegó a un valor máximo de 3.6516 x $10^{33}cm^{-2}s^{-1}$ con hasta 15 interacciones de media por cada cruce de haces y con un espacio mínimo de 50 ns entre haces. La frecuencia de *Trigger* de primer nivel superó los 50 kHz, lo que forzó un cambio en el modo de operación de los RODs en TileCal. Durante este año únicamente las muestras de canales con una diferencia entre la muestra máxima y mínima por encima de 5 cuentas de ADC eran transmitidas por el ROD. Además, el resultado de la reconstrucción se transmitía para todos los canales. Con esto, se reducía el tamaño de los paquetes de datos de salida del ROD y así se reducía el tiempo muerto introducido. La reconstrucción proporcionada por los RODs era utilizada en los algoritmos del *High Level Trigger* (HLT). Para los análisis de física se realizaba una reconstrucción de la energía *offline* para aquellos canales de los que se disponían las muestras, y la reconstrucción proporcionada por los RODs para el resto de canales.

Durante 2011 la luminosidad total integrada producida por el LHC en ATLAS llegó a los



(a) (b)

Figura 9.11: Tamaño del fragmento de salida del ROD en número de palabras de 32-bits para cada vínculo de salida de todos los módulos ROD con el formato de 2011 (a) y de 2012 (b) para un mismo conjunto de datos.

5.61 fb^{-1} con 5.25 fb^{-1} recogida en ATLAS, con una eficiencia total de adquisición de 93,6 %. De nuevo, TileCal tuvo una contribución a la ineficiencia total de ATLAS similar al resto de subdetectores y del 13,61 %. Este tiempo muerto en TileCal se produjo principalmente por el tiempo de espera de los RODs cuando la electrónica del detector cesaba la transmisión de datos mientras su fuente de alimentación tenía que ser restablecida por un fallo durante la toma de datos. Además, durante este año el sistema de *back-end* de TileCal no produjo una parada de un *run* mientras se producían colisiones en el detector.

El LHC continuó evolucionando y acercándose a los valores de diseño en 2012. La energía en centro de masas de incrementó hasta los 8 TeV y la luminosidad instantánea máxima llegó a los 7.31 x 10^{33} cm^{-2}s^{-1}, cercana a los 10^{34} cm^{-2}s^{-1} de diseño. Esta configuración permitió al LHC entregar un total de 23.3 fb^{-1} de los cuales 21.7 fb^{-1} fueron guardados por ATLAS. Los RODs sufrieron nuevas actualizaciones en el firmware principalmente con el fin obtener la máxima eficiencia de adquisición con el aumento de la frecuencia de *Trigger* de primer nivel por encima de los 70 kHz. Para ello, se implementó un algoritmo de compresión de datos con el fin de reducir el tamaño del fragmento de salida del ROD, principal causa de tiempo muerto a finales de 2011.

La Figura 9.11 muestra el tamaño del fragmento por cada enlace de salida de los RODs para el formato de datos utilizado en 2011 (a) y en 2012 (b). Dos lineas verticales muestran

el límite para funcionar a una frecuencia de *Trigger* de primer nivel de 100 kHz y 75 kHz sin introducir tiempo muerto. La contribucion del sistema de *back-end* de TileCal a la ineficiencia de adquisición de ATLAS durante 2012 fue del 9.9 %, valor similar al resto de subdetectores.

9.3 Algoritmos de Reconstrucción de Señal en los Módulos ROD de TileCal

Las muestras digitales correspondientes a los PMTs de TileCal son transmitidas a los módulos ROD para los eventos seleccionados por el primer nivel de *Trigger*. Con el fin de reconstruir la amplitud y el tiempo de la señal generada en los PMTs, varios algoritmos de reconstrucción de señal fueron evaluados y finalmente se seleccionó el denominado *Optimal Filtering* por ser el más adecuado a la arquitectura de computación de los DSPs disponibles en los RODs y debido al reducido tiempo de procesado disponible. Los algoritmos de reconstrucción proporcionan la energía calibrada, el tiempo de la deposición, un factor de calidad de la reconstrucción y la suma total de energía para cada módulo del detector así como sus proyecciones transversa y en Z en un fragmento dedicado. Este fragmento es utilizado en el HLT para realizar un *Trigger* de E_T^{miss}.

9.3.1 El Método *Optimal Filtering*

Optimal Filtering es un filtro digital el cual precisa un conocimiento previo de la forma de pulso con el fin de reducir la contribución del ruido electrónico y para determinar el tiempo de la deposición. El pedestal se define como la base de la señal mientras que el tiempo reconstruido como la diferencia temporal entre el pico del pulso y el tiempo de llegada esperado (según calibración) (Figura 9.12).

La amplitud es el parámetro de mayor interés, ya que es proporcional a la energía depositada en la celda, y se define como la altura del pulso medida desde el pedestal. En el caso de TileCal hay dos fuentes de ruido que pueden distorsionar el pulso, produciendo un error en el resultado de la reconstrucción. En primer lugar el ruido térmico intrínseco a la electrónica y que produce pequeñas variaciones de las muestras sobre el valor esperado para la forma de

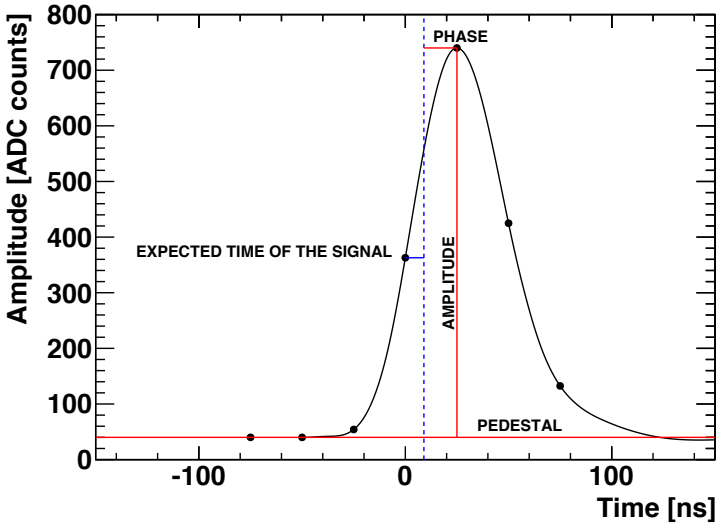


Figura 9.12: Forma del pulso con la definición de la amplitud, tiempo y pedestal. Los puntos representan las 7 muestras transmitidas del detector a los RODs para cada suceso y PMT.

pulso conocida. Por otro lado, el ruido por apilamiento lo causan deposiciones de energía en una misma celda por partículas producidas en distintas colisiones, ya sea en el mismo o en distinto pero cercano en el tiempo cruce de haces.

La Ecuación (9.1) representa la formulación matemática del método *Optimal Filtering* para calcular la Amplitud (A) y el Tiempo (τ) de una señal.

$$A = \sum_{i=1}^{n} a_i (S_i - p) \ , \qquad A\tau = \sum_{i=1}^{n} b_i (S_i) \tag{9.1}$$

Así mismo, la definición del Factor de Calidad (QF, *Quality Factor*) de la reconstrucción se expresa como:

$$Q_{\mathrm{F}} = \sqrt{\sum_{i=1}^{n} \left(S_i - (Ag_i + A\tau g_i' + p)\right)^2} \tag{9.2}$$

El QF proporciona una estimación de la diferencia entre las muestras recibidas y las

ideales para el resultado obtenido de la amplitud y tiempo asumiendo una forma de pulso ideal. Cualquier distorsión en la forma del pulso, por ejemplo producida por ruido electrónico o de apilamiento, causa un aumento en la diferencia entre muestras reales y las ideales y por tanto en el QF, que puede utilizarse para estimar la exactitud de la reconstrucción.

9.3.2 Implementación de OF en los Procesadores Digitales de Señal (DSP) de los Módulos ROD

La implementación de *Optimal Filtering* en la DSP se realizó con el fín de optimizar su cálculo y adaptarlo a su arquitectura de computación.

$$A = \sum_{i=1}^{7} S_i a_i \qquad \tau = \frac{1}{A}\sum_{i=1}^{7} S_i b_i \qquad QF = \sqrt{\frac{1}{512}\sum_{i=1}^{7} S_i - Ag_i + A\tau g'_i - ped} \tag{9.3}$$

Los pesos utilizados para el cálculo de la amplitud (a_i), del tiempo (b_i) y del QF (g_i, q'_i), y las constantes de calibración son copiados a la memoria de la DSP en tiempo de configuración a través del bus VME. La organización de los pesos y de los datos de entrada en la memoria de la DSP se realiza con el fin de optimizar el cálculo de los algoritmos.

En primer lugar, se calcula la amplitud y el tiempo del pulso para los 48 canales correspondientes a cada módulo del detector (Figura 9.13). La división por la amplitud, necesaria para estimar el tiempo según la Ecuación (9.3), se realiza mediante el método de *Look-Up-Table*, consistente en el almacenamiento en la memoria de la DSP del valor de 1/A para unos valores de la amplitud en todo el rango. Finalmente, la amplitud es calibrada a unidades de energía mediante una multiplicación de la amplitud por la constante de calibración de cada canal.

A continuación se calcula la suma total de la energía para el módulo, considerando únicamente los canales con una señal reconstruida por encima de un umbral, con el fin de minimizar la contribución del ruido electrónico. Asímismo, se calcula la proyección transversa y en Z de la energía total, de acuerdo a las coordenadas correspondientes del módulo procesado. En el caso de un tipo de *Trigger* de baja frecuencia (*Zero Bias*), las muestras digitales de todos los canales son transmitidas por la DSP y utilizadas para estudiar el ruido del detector. Para

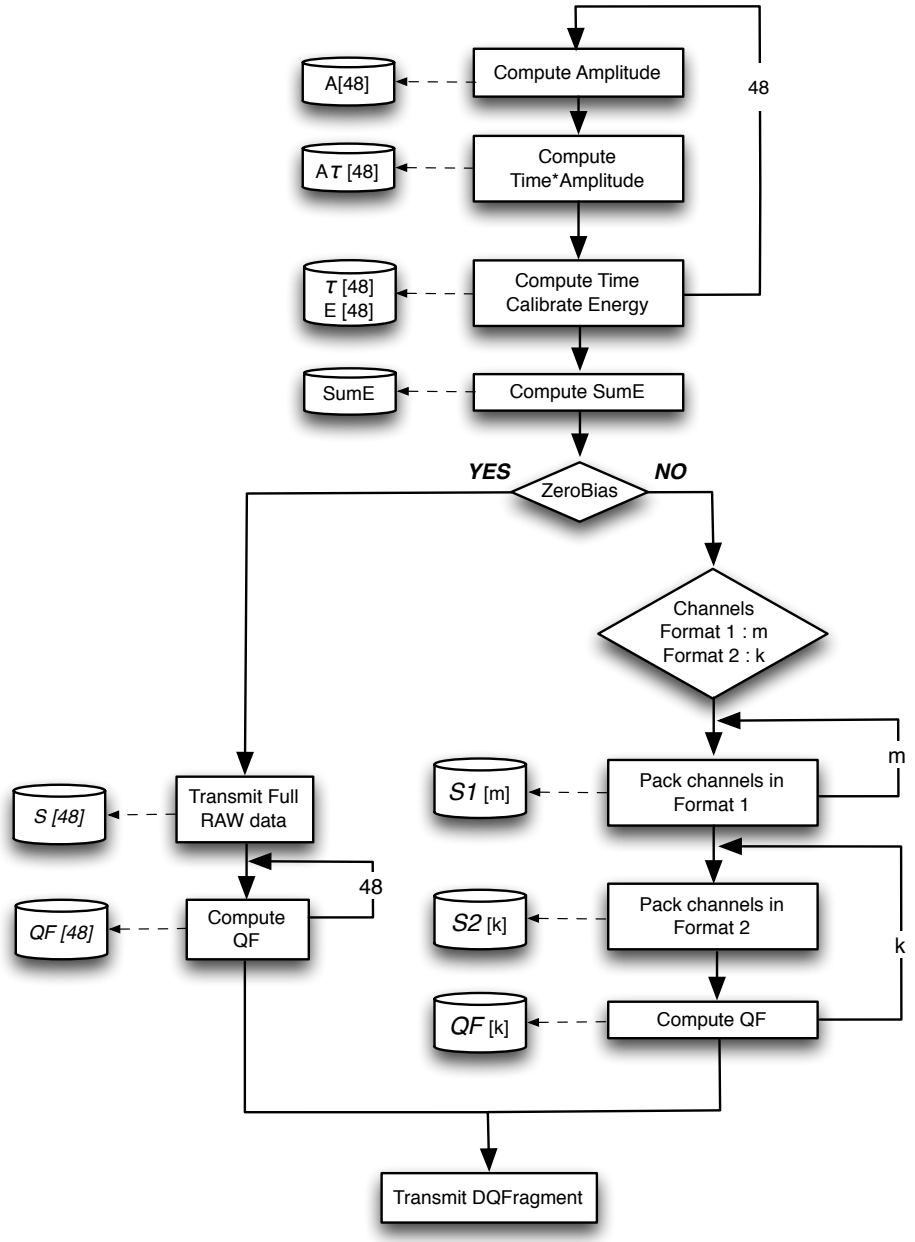


Figura 9.13: DSP reconstruction algorithm code flow.

eventos de física, únicamente las muestras de canales con una diferencia entre la muestra máxima y mínima por encima de un umbral configurable (por defecto de 5 cuentas de ADC) son transmitidas. Asímismo, para canales con esa diferencia por encima de 16 cuentas de ADC se cálcula el QF, mientras que para el resto de canales se fuerza a cero puesto que el pequeño valor de las muestras no proporcionan un resultado diferente de cero en ningún caso. Debido a que únicamente se disponen de 4 bits en el formato de datos de salida para codificar el valor estimado del QF, el resultado es escalado (Ecuación (9.3)) con el fin de cubrir un mayor rango de valores a costa de reducir su precisión.

Desviación parabólica y corrección Los pesos utilizados por *Optimal Filtering* son dependientes de la forma del pulso, que se considera constante para todos los canales, y de el tiempo esperado de la señal, que se mide como el tiempo entre la muestra central y el máximo de un pulso generado por una partícula proveniente del punto de interacción. La electrónica del detector se calibra para conseguir que el máximo de cada señal coincida con el tiempo esperado. En ese caso, el tiempo reconstruido por *Optimal Filtering* es cero y la amplitud corresponde al máximo del pulso, según la Figura 9.12. La reconstrucción de la amplitud para señales con el máximo distinto al esperado se degrada y se desvía de la amplitud real del pulso. Este efecto puede ser utilizado para minimizar la contribución de deposiciones de energía producidas en cruces de haces distintos al evento de interés, ya que producen señales con tiempos distintos al esperado. No obstante, la desviación de la amplitud reconstruida respecto a la amplitud real de la señal puede ser parametrizada en función del tiempo de la señal, y posteriormente corregida si se considera de interés. La Figura 9.14 muestra la diferencia relativa entre la amplitud reconstruida y la inyectada en función del tiempo reconstruido sin aplicar la corrección (rojo) y tras ser aplicada (gris). Esta corrección es aplicada para variaciones de tiempo pequeñas y que pueden deberse, por ejemplo, a partículas de distinto momento.

Método iterativo Con el fin de disminuir al máximo la desviación de la amplitud reconstruida con el tiempo, se puede utilizar el método *Optimal Filtering* de manera iterativa. En la primera iteración se calcula el tiempo de la señal utilizando los pesos asumiendo un tiempo esperado igual a cero. En la segunda y tercera iteraciones se utilizan los pesos correspondientes al tiempo reconstruido en la iteración previa. La tercera y última iteración proporciona la

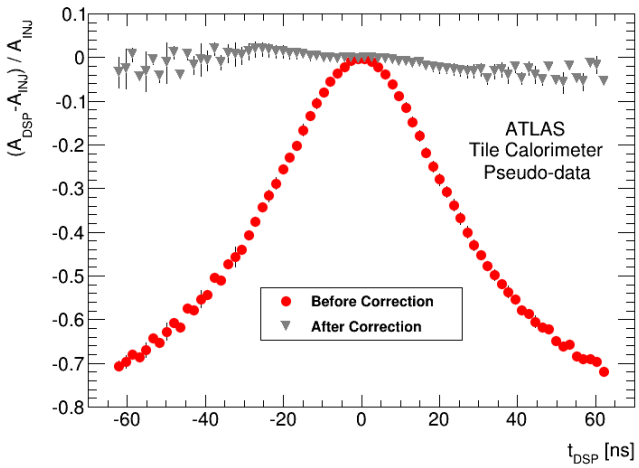


Figura 9.14: Diferencia relativa entre la amplitud inyectada y reconstruida en función del tiempo reconstruido antes (círculos rojos) y después (triangulos gises) de aplicar la corrección parabólica.

amplitud reconstruida calculada con los pesos más apropiados a la señal recibida. El tiempo de procesado de este método triplica el del método no iterativo y puede ser utilizado únicamente para frecuencias de *Trigger* de primer nivel por debajo de 10 kHz. Así mismo, este método puede ser utilizado cuando el espacio entre haces en el anillo supera los 75 ns, tiempo entre la muestra central y la primera o última del pulso digital. Para espacios entre haces menores, el algoritmo reconstruye la amplitud del pulso máximo presente en las 7 muestras, que puede deberse a un cruce anterior o posterior al seleccionado que se encuentra en la muestra central.

9.4 Validación y Funcionamiento de los Métodos de Reconstrucción de Señal

Durante la puesta a punto del detector el procesamiento de señal en los RODs fue utilizado con datos de calibración y en toma de datos de muones cósmicos. En ambos casos, la frecuencia de *Trigger* de primer nivel se encontraba por debajo de 10 kHz y se utilizó el método iterativo por defecto. Además, este método es especialmente óptimo para la reconstrucción de señales

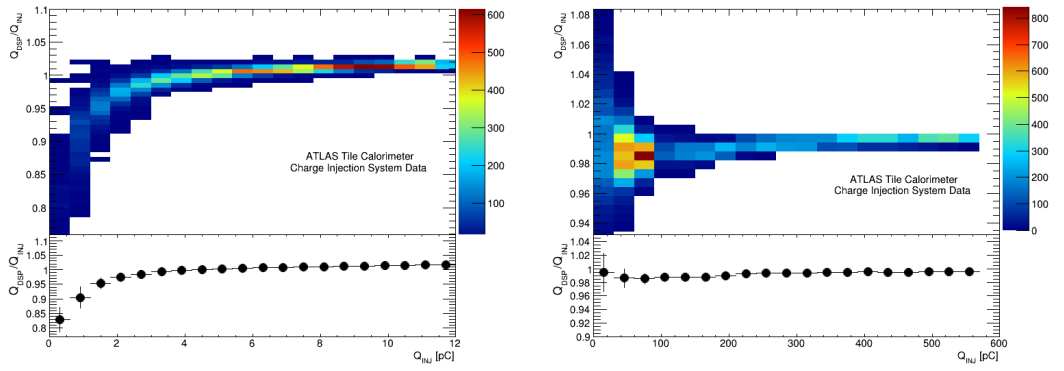
producidas por muones cósmicos con un tiempo de paso por el detector aleatorio. El método iterativo se utilizó por defecto tanto en los RODs como en la reconstrucción *offline* durante la puesta a punto del detector, así como durante las primeras colisiones, donde las frecuencias de *Trigger* de primer nivel fueron bajas y las distancias entre haces por encima de 75 ns. Esos primeros datos fueron utilizados para calibrar temporalmente el detector mediante el cálculo del tiempo esperado para cada canal. Con el aumento de la frecuencia de *Trigger* de primer nivel y la reducción de espacio entre haces a 50 ns, el algoritmo no iterativo se convirtió en el método de reconstrucción por defecto tanto en los RODs como *offline*.

9.4.1 Evaluación de la Reconstrucción de Señal con Datos de Calibración

El sistema de inyección de carga (CIS, *Charge Injection System*) ha sido utilizado para evaluar la calidad de la reconstrucción de señal en las DSPs utilizando datos reales producidos en la electrónica del detector controlando la amplitud de las señales procesadas. Estas señales incluyen ruido electrónico así como una forma de pulso distinta a las producidas por deposiciones de energía de partículas atravesando el detector. En particular, la forma del pulso de CIS incluye una contribución constante negativa, conocida como *leakage pulse*, producida trás el pulso y posteriormente se restaura el valor inicial del pedestal. El análisis se realizó utilizando un único canal del detector con el fin de minimizar efectos no producidos por la reconstrucción de señal.

La Figura 9.15 muestra el ratio entre la carga reconstruida por la DSP y la carga teórica inyectada en función de la carga inyectada. Como se ha dicho, la forma del pulso del sistema de inyección de carga incluye un término que lo distorsiona y que al ser constante con la carga, tiene un efecto más pronunciado a cargas menores, y principalmente en el caso de alta ganancia. Los resultados muestran que la carga reconstruida por la DSP y la inyectada son compatibles con diferencias menores al 1 % para cargas por encima de 7 pC en alta ganancia y de 100 pC en baja ganancia.

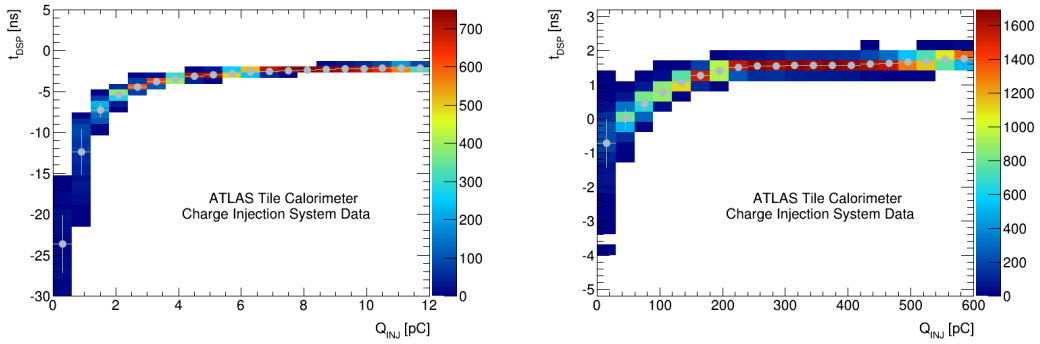
El sistema de inyección de carga fue configurado para inyectar pulsos de distinta amplitud pero con tiempo constante y absoluto igual a cero, es decir, con el máximo del pulso en la muestra central. La Figura 9.16 muestra el tiempo reconstruido por la DSP en función de



(a) (b)

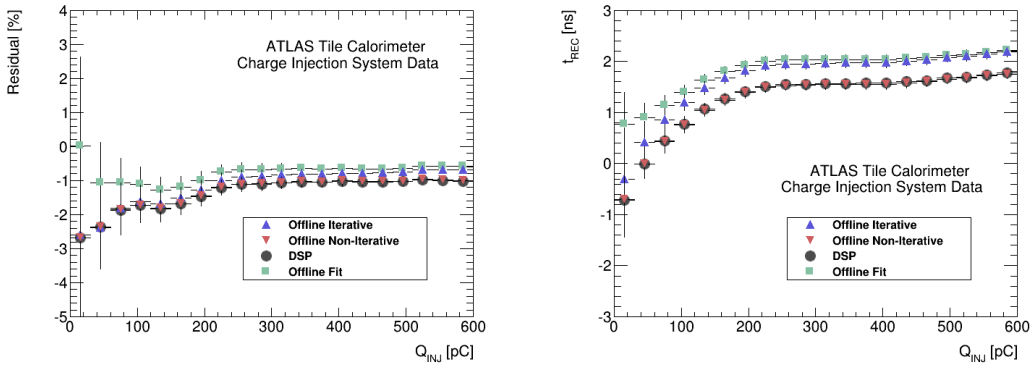
Figura 9.15: Ratio entre la carga inyectada y reconstruida en función de la carga inyectada para alta (a) y baja (b) ganancias utilizando el sistema de inyección de carga. En la parte inferior se muestra el valor medio y el RMS.

la carga inyectada. De nuevo, para cargas situadas en la primera mitad de cada ganancia el efecto del *leakage pulse* es más pronunciado, mientras que para cargas mayores el tiempo se estabiliza. La diferencia con cero es el residuo y puede ser utilizado como tiempo esperado para calcular los pesos de *Optimal Filtering* y obtener un tiempo reconstruido igual a cero.



(a) (b)

Figura 9.16: Tiempo reconstruido por la DSP en función de la carga inyectada para alta (a) y baja (b) ganancia con el sistema de inyección de carga. Los puntos grises muestran la media y el RMS.



(a) (b)

Figura 9.17: Diferencia relativa entre la carga inyectada y reconstruida (a) y tiempo reconstruido (b) en ambos casos en función de la carga inyectada y en baja ganancia. Se muestran los resultados para la reconstrucción con *Optimal Filtering* en la DSP (círculos negros), *Optimal Filtering* con coma fija (triángulos rojos), *Optimal Filtering* iterativo (triángulos azules) y el método *FIT* (cuadrados verdes).

Además, podemos validar la reconstrucción en la DSP comparando los resultados con los obtenidos con otros métodos de reconstrucción de señal. Utilizando las muestras digitales, aplicamos la reconstrucción en la infraestructura de software de ATLAS (Athena) y en procesadores de coma flotante, con el fin de evaluar el impacto de utilizar una aritmética de coma fija en la DSP. Se utiliza como referencia una versión en coma flotante de *Optimal Filtering*, *Optimal Filtering* iterativo, y el método *FIT*. La Figura 9.17a muestra la diferencia relativa entre la carga inyectada y la reconstruida (residuo) en función de la carga inyectada para los cuatro métodos en baja ganancia. Para cargas por debajo de 200 pC se observa una discrepancia del método *FIT* respecto al resto, debido a que en ese método se substrae de las muestras el *leakage pulse* previamente a realizar la reconstrucción. Para cargas por encima de los 200 pC, los cuatro métodos son equivalentes. La Figura 9.17b muestra el tiempo reconstruido en función de la carga inyectada en baja ganancia para los cuatro métodos de reconstrucción. De nuevo, los cuatro métodos muestran una compatibilidad por debajo de 1 ns.

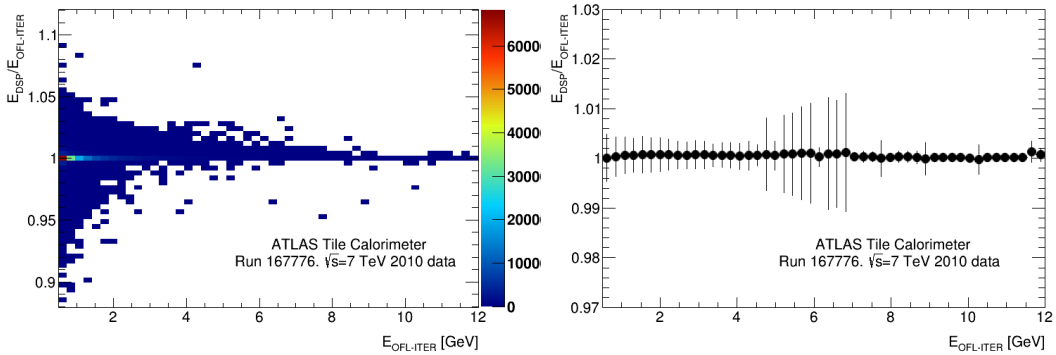


Figura 9.18: Ratio entre la energía reconstruida en la DSP y *offline* con el método iterativo en función de la energía *offline* para el *run* de física 167776 de 2010. La Figura de la derecha muestra la media y el RMS.

9.4.2 Evaluación con Datos de Física Producidos en Colisiones

Durante el año 2010 el LHC operó con un reducido número de haces y con una separación superior a 75 ns entre ellos. Esto permitió utilizar como algoritmo de reconstrucción *offline* el método *Optimal Filtering* iterativo, con el fin de reducir la desviación parabólica. Por otro lado, en la DSP se utilizó en método no iterativo debido al reducido tiempo de procesado disponible. Pese a que la corrección parabólica era aplicada sobre el resultado de la DSP con el fin de reducir la desviación, las diferencias entre el resultado proporcionado por la DSP y *offline* se encuentra dominado por los pulsos con tiempos grandes, como se muestra en la Figura 9.18. Para energías por debajo de 4 GeV la distribución temporal de las deposiciones es más dispersa y el efecto de la desviación parabólica más acusado.

Con el fin de minimizar el efecto de la desviación parabólica y caracterizar de manera más precisa la limitación de la reconstrucción en la DSP, se puede realizar una selección de pulsos con tiempo cercano a cero. La Figura 9.19 muestra que los resultados del algoritmo *offline* iterativo y de la DSP son compatibles para pulsos con un tiempo absoluto inferior a 1 ns.

Puesto que el resultado de la amplitud es utilizado para estimar el tiempo del pulso, la desviación parabólica también afecta en segundo orden a la reconstrucción del tiempo.

Para tiempos en el rango de [-12.5, 12.5] ns (correspondiente a la distancia mínima entre haces en los valores de diseño del LHC) los resultados de la DSP y *offline* con el método

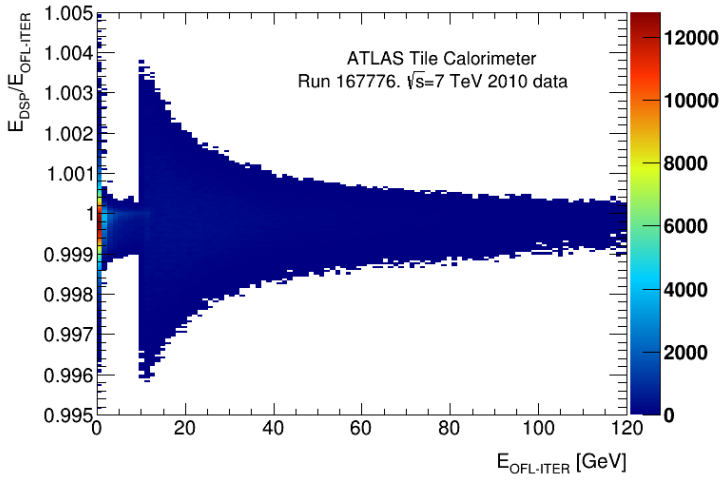


Figura 9.19: Ratio entre la energía reconstruida en la DSP y *offline* con el método iterativo en función de la energía *offline* para pulsos con un tiempo reconstruido en valor absoluto inferior a 1 ns para datos de física del año 2010.

iterativo son compatibles, y para tiempos mayores esta correlación se degrada (Figura 9.20). No obstante, como se observa en la Figura 9.20, el 95 % de las deposiciones de energía en datos de física se producen en tiempos en valor absoluto inferiores a 5 ns.

Con el objetivo de aumentar la luminosidad en el LHC, en 2011 se incrementó el número de haces en el anillo reduciéndose la distancia mínima entre ellos a 50 ns. Esto provocó un cambio en la estrategia de reconstrucción en TileCal, utilizándose el algoritmo *Optimal Filtering* no iterativo por defecto, con el fin de reducir el efecto del apilamiento procedente de interacciones distintas al evento seleccionado. Como se observa en la Figura 9.21, el método iterativo maximiza las deposiciones energéticas producidas en los cruces de haces anterior (+50 ns) y posterior (-50 ns) al evento seleccionado, mientras el algoritmo no iterativo minimiza su efecto.

Así pues, la única diferencia entre la reconstrucción en la DSP y *offline* para datos de 2011 y posteriores se debe a la limitación en la precisión de la aritmética de coma fija utilizada en la DSP respecto a los computadores de coma flotante utilizados para la reconstrucción *offline*.

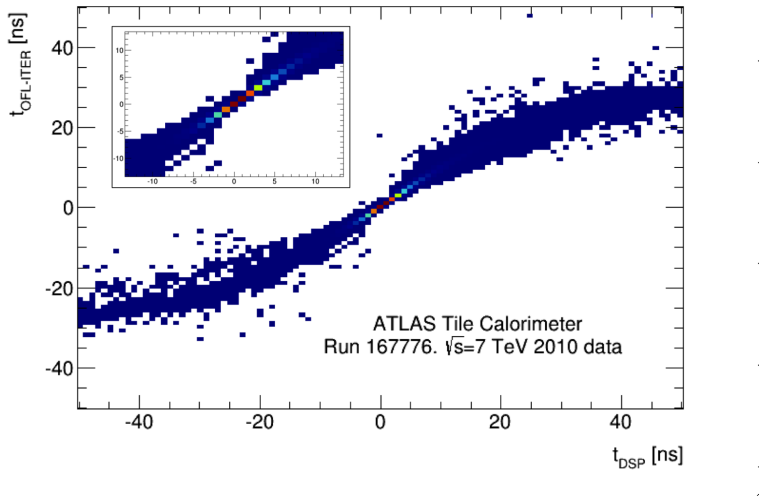
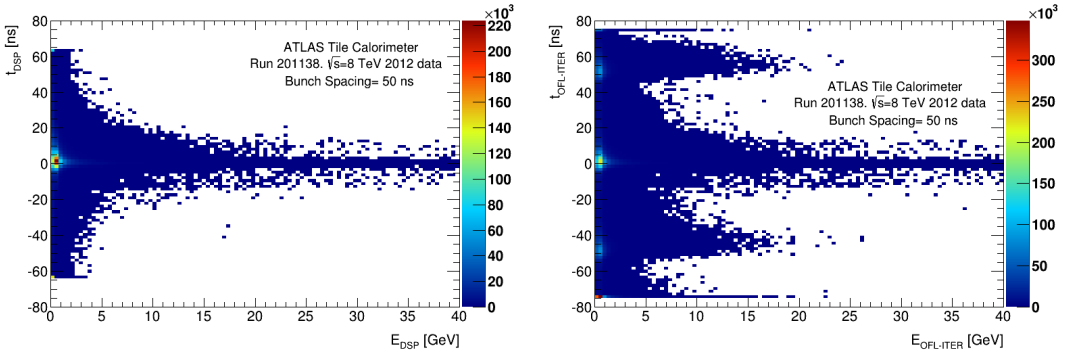


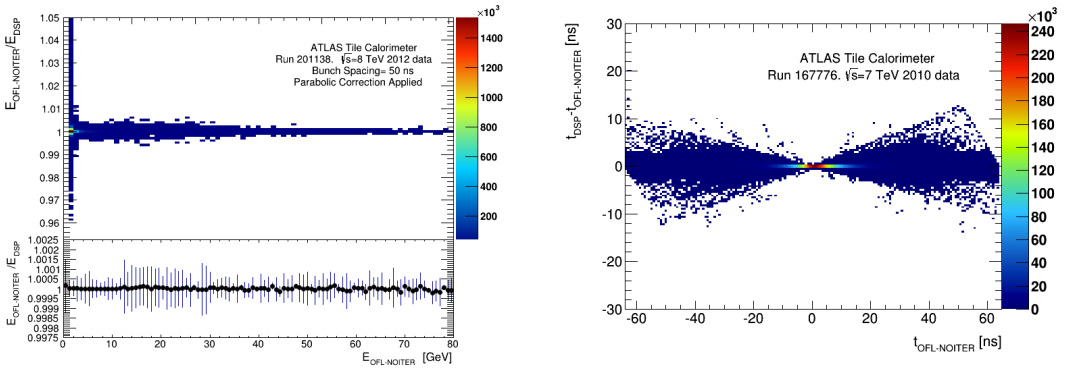
Figura 9.20: Correlación entre el tiempo reconstruido en la DSP y *offline* con el método iterativo.



(a) (b)

Figura 9.21: Distribución del tiempo en función de la energía reconstruidos con la DSP (a) y *offline* con el método iterativo (b) para el *run* 201138 de física con una distancia mínima entre haces de 50 ns.

La Figura 9.22a muestra el ratio entre la energía reconstruida en la DSP y *offline* para el *run* 201138 de 2012 con el método por *Optimal Filtering* no iterativo en ambos casos. Los resultados son compatibles por debajo del 1 % en todo el rango de energías.



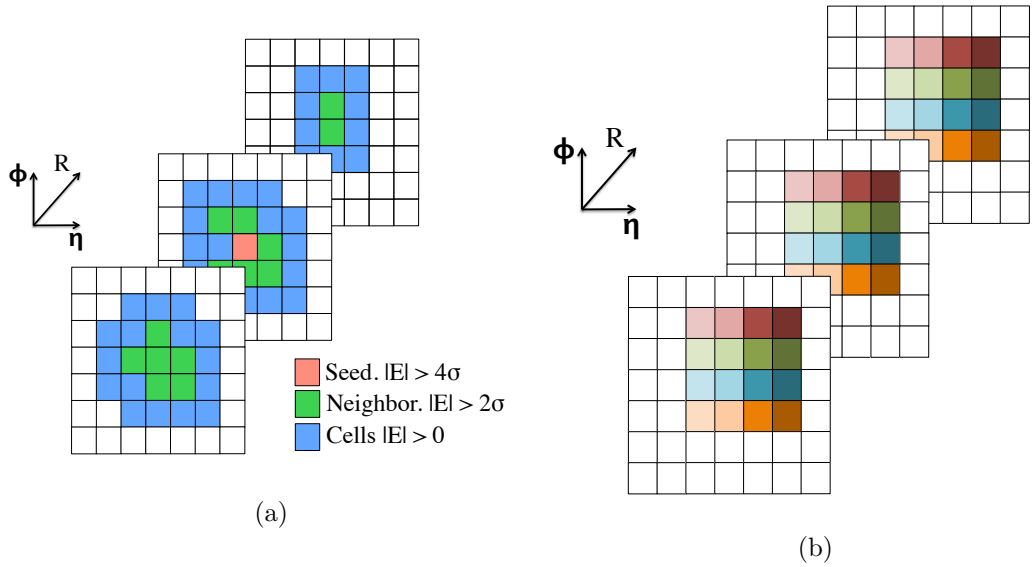
(a) (b)

Figura 9.22: a) Ratio entre la energía reconstruida en la DSP y *offline* con el método no iterativo en función del tiempo *offline*. b) Diferencia absoluta entre el tiempo reconstruido en la DSP y *offline* con el método no iterativo en función del tiempo *offline*. El efecto introducido al realizar la división por la energía con una LUT es proporcional al valor absoluto del tiempo. En ambas Figuras se utilizan datos del *run* 201138 de 2012.

La mayor limitación en la reconstrucción del tiempo en la DSP se produce por la utilización de la LUT para implementar la división por la amplitud. Además, el error introducido por la LUT es proporcional al valor absoluto del tiempo reconstruido. La Figura 9.22b muestra la diferencia absoluta entre el tiempo reconstruido en la DSP y *offline* con el método no iterativo en función del tiempo *offline* donde se observa que las diferencias en valor absoluto entre ambos métodos pueden llegar a los 10 ns para tiempos al final del rango. No obstante, para tiempos considerados al cruce de haces del evento seleccionado, rango [-12.5, 12.5] ns, y donde se sitúan más del 98 % de las deposiciones, la diferencia entre ambos resultados se encuentran por debajo de 3 ns.

Estudio de la Reconstrucción con *Jets*

Una vez evaluada la precisión de la arquitectura de coma fija utilizada en la DSP en la reconstrucción de señales, se desea estudiar su impacto en la reconstrucción de objetos físicos más complejos como son los *jets*. El objetivo es cuantizar el efecto en la reconstrucción de *jets* en ATLAS al utilizar el resultado proporcionado por la DSP en comparación a los métodos *offline* para la reconstrucción de señal en TileCal.



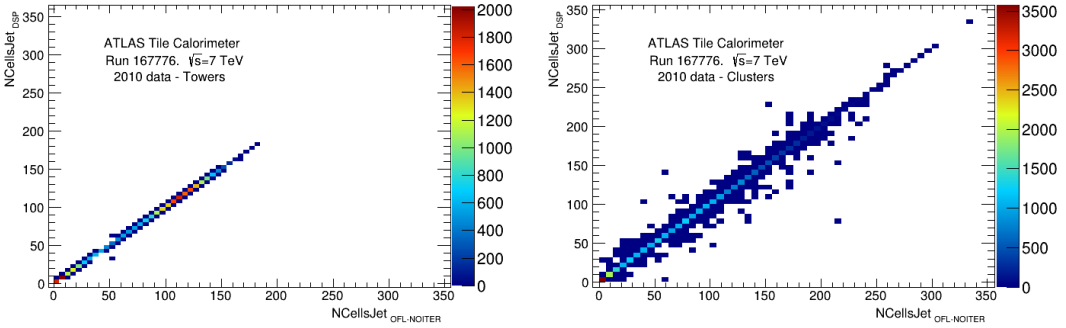
(a) (b)

Figura 9.23: Esquema de la estructura de un *cluster* (a) y una torre (b) en tres capas del calorimetro en un cono de $dR = 0{,}4$.

El algoritmo oficial de reconstrucción de *jets* en ATLAS es el denominado $anti - k_t$. La reconstrucción de un *jet* se realiza partiendo de una lista de constituyentes más básicos donde el momento del *jet* antes de aplicar ninguna corrección se define como:

$$P_{jet} = \sum P_{constituyentes}. \tag{9.4}$$

Los constituyentes o entradas para al algoritmo de reconstrucción de *jets* pueden ser torres calorimétricas o *cluster* tipológicos tridimensionales. Las torres calorimétricas incluyen todas las celdas del calorímetro en un área η x ϕ = 0.1 x 0.1 y por lo tanto presenta un tamaño o fijo (Figura 9.23). Los *clusters* topológicos se forman con el algoritmo llamado *4/2/0*, que comienza con una selección de celdas semilla con una energía $|E|>4\sigma$, donde σ corresponde al ruido característico de la celda (electrónico y de apilamiento). El *cluster* también incluye las celdas vecinas a las semillas y con una energía $|E|>2\sigma$ y finalmente se incluyen todas las celdas vecinas con $|E|>0$ (Figura 9.23). Por lo tanto, la estructura y tamaño de los *clusters* topológicos no son fijos, y depende de las deposiciones energéticas y del ruido característico de cada celda.



(a) (b)

Figura 9.24: Correlación entre el número de celdas dentro de un *jet* utilizando la DSP o el algoritmo *offline* no iterativo como método de reconstrucción en TileCal, utilizando torres (a) o *clusters* (b) como constituyentes básicos de los *jets*.

Durante el estudio se realiza la reconstrucción de un mismo grupo de datos cambiando únicamente el algoritmo de reconstrucción para los canales de TileCal, lo que permite realizar una comparación evento por evento. Dentro de cada evento, se seleccionan los *jets* que contengan al menos una traza asociada en el detector interno de ATLAS y al menos contengan una celda en TileCal. Finalmente, se seleccionan *jets* aislados (no existe otro jet en un cono de radio R<0.4) y que el *jet* esté presente en ambos métodos de reconstrucción. Finalmente, únicamente el *jet* más energético por evento que cumpla la selección es utilizado para las comparaciones.

El efecto producido al utilizar torres o *clusters* como constituyentes básicos de los *jets* se observa en la Figura 9.24, donde se muestra la correlación entre el número de celdas dentro del *jet* utilizando la reconstrucción de la DSP o el algoritmo *offline* no iterativo como método de reconstrucción en TileCal. La topología homogénea de las torres produce una mejor correlación entre los métodos, mientras que la estructura de los *clusters* se ve alterada por cualquier variación de la energía reconstruida y empeora la correlación entre los métodos.

Otro efecto importante en la reconstrucción de *jets* es el apilamiento, que es proporcional a la luminosidad instantánea y al número de interacciones por cruce de haces en el LHC. Para datos de física del año 2010, donde la luminosidad se puede considerar baja respecto a los valores de diseño y con menos de una iteración por cruce de haces de media, la diferencia en la

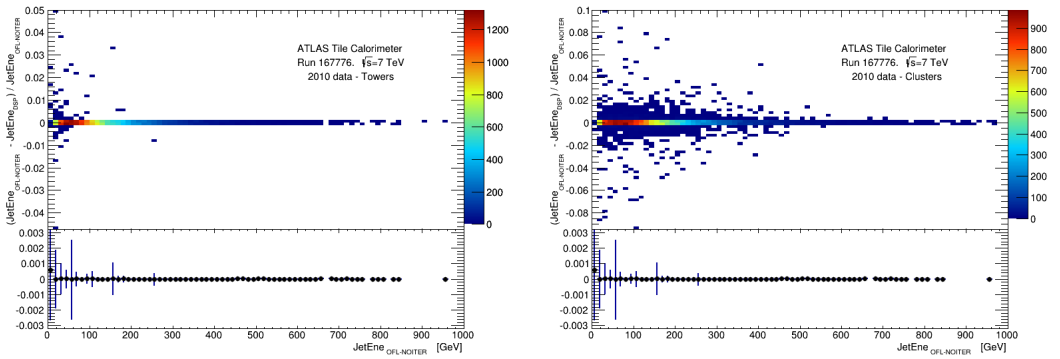


Figura 9.25: Diferencia relativa en la energía total de los *jets* entre utilizar la DSP o el algoritmo *offline* no iterativo como método de reconstruction de señal en TileCal, y usando torres (a) y *clusters* (b) como parámetro de entrada en el algoritmo de reconstrucción de *jets* para el *run* de física 167776 de 2010. En la parte inferior se muestra la media y el RMS.

energía total de los *jets* reconstruidos utilizando el resultado de la DSP o el algoritmo *offline* no iterativo como método de reconstrucción en TileCal se encuentra por debajo del 1 % en valor medio (Figura 9.25). No obstante, las diferencias son menores en el caso de utilizar torres frente a *clusters*, debido a la diferente topología de los *jets* formados a partir de *clusters*.

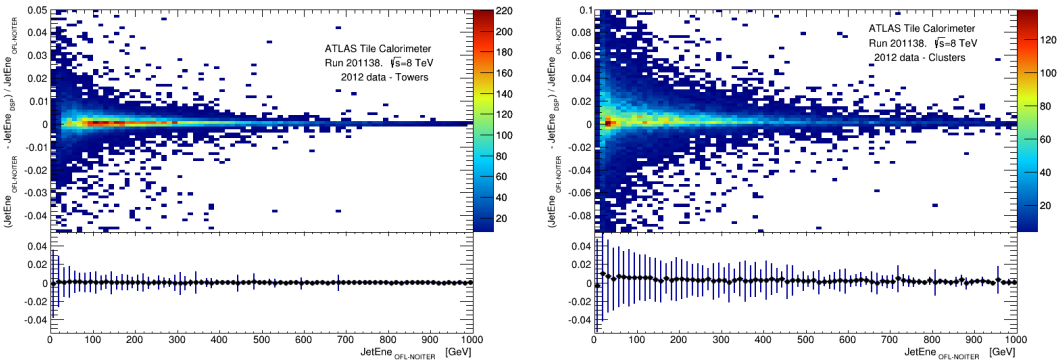


Figura 9.26: Diferencia relativa en la energía total de los *jets* entre utilizar la DSP o el algoritmo *offline* no iterativo como método de reconstruction de señal en TileCal, y usando torres (a) y *clusters* (b) como parámetro de entrada en el algoritmo de reconstrucción de *jets* para el *run* de física 201138 de 2012. En la parte inferior se muestra la media y el RMS.

En 2012 el LHC aumentó considerablemente su luminosidad instantánea, y con más de

30 interacciones por cruce de haces. Esto produjo un aumento considerable del apilamiento afectando al funcionamiento de la reconstrucción de *jets* en ATLAS. Este efecto se observa al comparar la energía total de los *jets* reconstruidos utilizando la DSP o el algoritmo *offline* no iterativo como método de reconstrucción en TileCal. La diferencia en la energía total del *jet* entre ambos métodos aumenta para el caso de alta luminosidad (Figura 9.26) respecto al caso de baja luminosidad (Figura 9.25) debido al apilamiento de señales. La diferencia entre ambos métodos mejora con la utilización de torres como constituyentes de los *jets* frente a *clusters*, y en ambos casos el valor medio de la diferencia entre métodos se encuentra cercana a cero con valores de RMS inferiores al 5 %.

Acronyms

ADC	Analog to Digital Converter
ALICE	A Large Ion Collider Experiment
ALU	Arithmetic Logic Unit
ASIC	Application Specific Integrated Circuit
ATLAS	A Toroidal LHC ApparatuS
BCID	Bunch Crossing IDentifier
BCR	Bunch Crossing Reset
BER	Bit Error Rate
CAN	Controller Area Network
CERN	Conseil Europen pour la Recherche Nucleaire
CIS	Charge Injection System
CMS	Compact Muon Solenoid
CP	Cluster Processor
CSC	Cathode Strip Chamber
CTP	Central Trigger Processor
DAC	Digital to Analog Converter
DCS	Detector Control System
DFM	DataFlow Manager
DMA	Direct Memory Access
DMU	Data Management Unit
DSP	Digital Signal Processor
DVS	Diagnostic and Verification System

E_T^{miss}	Missing Transverse Energy
EB	Extended Barrel
ECR	Event Counter Reset
ECRID	Event Counter Reset IDentifier
EF	Event Filter
EMEC	ElectroMagnetic EndCap Calorimeter
EMIF	External Memory Interface
EVID	Event IDentificator
FCal	Forward Calorimeter
FIFO	First In First Out
FPGA	Field Programmable Gate Array
FSM	Finite State Machine
HEC	Hadronic EndCap Calorimeter
HLT	High Level Trigger
HOLA	High Speed Optical Link
HPI	Host Ports Interface
IGUI	Integrated Graphical User Interface
IP	Interaction Point
IPC	Inter Process Communication
IS	Information Services
JEP	Jet/Energy-sum Processor
JTAG	Joint Test Action Group
L1A	Level 1 Accept
L1Calo	Level 1 Calorimeter
L2PU	Level 2 Processing Units
L2SV	Level 2 SuperVisor
LB	Long Barrel
LHC	Large Hadron Collider
LSB	Least Significant Bit
LTP	Local Trigger Processor

LTPI	LTP interface
LUT	Look Up Table
LVDS	Low Voltage Differential Signaling
MAC	Multiply ACcumulative
MB	Minimum Bias
McBSP	Multi channel Buffered Serial Ports
MDT	Monitored Drift Tube
MSB	Most Significant Bit
NIM	Nuclear Instrumentation Module
OB	Optical Buffer
OC	Output Controller
OFC	Optimal Filtering Constants
OMB	Optical Multiplexer Board
ORx	Optical Receiver
PCB	Printed Circuit Board
PECL	Positive Emitter Coupled Logic
PLL	Phase-Locked Loop
PMC	PCI Mezzanine Card
PMT	PhotoMultiplier Tube
PU	Processing Unit
RCC	ROD Crate Controller
RCD	ROD Crate DAQ
RIMBO	ROD Injector and Multiplexer BOard
ROB	ReadOut Buffer
ROD	ReadOut Driver
RoI	Region of Interest
ROL	ReadOut Link
RPC	Resistive Plate Chamber
SBC	Single Board Computer
SCT	SemiConductor Tracker

SEE	Single Event Error
SEU	Single Event Upsets
SFI	Sub-Farm Input
SFO	Sub-Farm Output
TBM	Trigger and Busy Module
TDAQ	Trigger and Data AcQuisition
TGC	Thin Gap Chamber
TID	Total Ionizing Dose
TileCal	Tile Calorimeter
TM	Transition Module
TRT	Transition Radiation Tracker
TTC	Trigger, Timing and Control
TTCex	TTC Emitter
TTCoc	TTC Optical Coupler
TTCvi	TTC bus Interface
TTL	Transistor Transistor Logic
TType	Trigger Type
VCXO	Voltage Controlled Oscillator
VLIW	Very Long Instruction Words
VME	Versa Module Eurocard
WLS	WaveLength-Shifting (WLS)

Bibliography

[1] The LHC Study Group. *The Large Hadron Collider Accelerator project.* CERN, Geneva, 1993. CERN AC/93-03-(LHC).

[2] ATLAS Collaboration. The ATLAS Experiment at the CERN Large Hadron Collider. *Nucl. Instrum. Meth.*, S08003(3), 2008.

[3] CMS Collaboration. *Technical Proposal.* CERN, Geneva, 1994. CERN-LHCC-94-38 ; LHCC-P-1.

[4] LHCb Collaboration. *LHCb: Technical Proposal.* CERN, Geneva, 1998. CERN-LHCC-98-004; LHCC-P-4.

[5] ALICE Collaboration. *ALICE : Technical proposal for a Large Ion collider Experiment at the CERN LHC.* CERN, Geneva, 1995. CERN-LHCC-95-71; LHCC-P-3.

[6] ATLAS Collaboration. Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC. *Physics Letters B*, 716(1):1 – 29, 2012.

[7] ATLAS Collaboration. *ATLAS Inner Detector Technical Design Report.* Technical Design Report ATLAS. CERN, Geneva, 1997. CERN/LHCC 97-16.

[8] ATLAS Collaboration. *ATLAS Tile Calorimeter Detector Technical Design Report.* Technical Design Report ATLAS. CERN, Geneva, 1996. CERN/LHCC 96-42.

[9] ATLAS Collaboration. *ATLAS muon spectrometer: Technical Design Report.* Technical Design Report ATLAS. CERN, Geneva, 1997. distribution.

[10] ATLAS Collaboration. *ATLAS magnet system: Technical Design Report.* Technical Design Report ATLAS. CERN, Geneva, 1997.

[11] ATLAS Collaboration. *ATLAS high-level trigger, data-acquisition and controls: Technical Design Report.* CERN, Geneva, 2003. ATLAS-TDR-016; CERN-LHCC-2003-022.

[12] A. Barriuso Poy et al. The detector control system of the atlas experiment. *Journal of Instrumentation*, 3(05):P05006, 2008.

[13] ATLAS Collaboration. *ATLAS level-1 trigger: Technical Design Report.* Technical Design Report ATLAS. CERN, Geneva, 1998.

[14] M. Nordberg P. Jenni, M. Nessi and K. Smith. *ATLAS high-level trigger, data-acquisition and controls Technical Design Report.* Technical Design Report ATLAS. CERN, Geneva, 2003.

[15] ATLAS Collaboration. Readiness of the ATLAS Tile Calorimeter for LHC collisions. *Eur. Phys. J. C.*, 70(CERN-PH-EP-2010-024):1193–1236, Jul 2010.

[16] ATLAS Tile Calorimeter Collaboration. The Optical Instrumentation of the ATLAS Tile Calorimeter, Nov 2011.

[17] K. Anderson et al. Design of the front-end analog electronics for the ATLAS Tile Calorimeter. *Nucl. Instrum. Meth.*, A(551):469–476, 2005.

[18] A. Marchioro J. Christiansen and P. Moreira. TTCrx: an ASIC for timing, trigger and control distribution in LHC experiments, 1996.

[19] K. Anderson et al. ATLAS Tile Calorimeter Interface. In *8th Workshop on Electronics for LHC Experiments*, Colmar, France, September 2002.

[20] B.G. Taylor. TTC distribution for LHC detectors. *IEEE Transactions on Nuclear Science*, 45:821–828, 1998.

[21] P. Borrego-Amaral et al. The ATLAS Local Trigger Processor (LTP) . Technical Report CERN-ATL-COM-DAQ-2004-025, CERN, Geneva, 2004.

[22] P. Gällnö. Modules development for the TTC system, Oct 1999.

[23] Van der Bij et al. S-link, a data link interface specification for the lhc era. *IEEE Trans. Nucl. Sci.*, 44:398–402, 1997.

[24] A. Kugel et al. A Robin Prototype for a PCI-Bus based ATLAS Readout-System. Technical Report ATL-COM-DAQ-2004-001, CERN, Geneva, Oct 2003.

[25] P. Matricon. The Trigger and Busy Module of the ATLAS LARG ROD System. ATLAS EDMS document ATL-AL-EN-0054.

[26] P. Matricon. A 9U Transition Module for the ROD Demonstrator. ATLAS EDMS document ATL-AL-EN-0053.

[27] K. Anderson et al. SEE Test of the TileCal Optical Interface Board, 2001.

[28] R. Blair et al. The atlas high level trigger region of interest builder. *Journal of Instrumentation*, 3(04):P04001, 2008.

[29] T. Liu. Optical links for atlas liquid argon calorimeter front-end electronics readout. Technical Report ATL-LARG-SLIDE-2010-301, CERN, Geneva, Sep 2010.

[30] Inc. Agilent Technologies. Agilent HDMP-1032A/1034A Transmitter/Receive Chip Set, August 16, 2001. Datasheet.

[31] A. Valero et al. Temperature studies of the tilecal rod g-links for the validation of the air-cooling system. Technical Report ATL-TILECAL-PUB-2007-003, CERN, Geneva, Mar 2007.

[32] Texas Instruments. *TMS320C6000 CPU and instruction set reference guide*, 2000. SPRU189F.

[33] A. Valero et al. DSP Online algorithms for the ATLAS TileCal Read-Out Drivers. *Nuclear Science, IEEE Transactions on Nuclear Science*, 55(1):158–164, February 2008.

[34] C. A. Solans. *Implementation of the ROD Crate DAQ Software for the ATLAS Tile Calorimeter and a Search for a MSSM Higgs Boson decaying into Tau pairs.* PhD thesis, Universitat de Valencia, Valencia, 2010.

[35] G. L. Miotto. Operations at different activity stages. Technical Report ATLAS-TDAQ-CONTROLS-2005-001, CERN, Geneva, 2005.

[36] C. Bee et al. The raw event format in the atlas trigger/daq. Technical Report ATL-DAQ-98-129, CERN, Geneva, Oct 1998.

[37] V. Gonzalez et al. Development of the Optical Multiplexer Board Prototype for Data Acquisition in the TileCal System. *Nuclear Science, IEEE Transactions on Nuclear Science*, 53(1):2131 – 2138, August 2006.

[38] Stratos Lightwave Optoelectronic. Optical Gigabit Ethernet-+3.3V Dual Small Form Factor (SFF) Receivers/Transmitters-1.25GBaud. Datasheet.

[39] Altera Corporation. Cyclone , 2003. Datasheet.

[40] A.Valero et al. The ATLAS tile calorimeter ROD injector and multiplexer board. *Nuclear Instruments and Methods in Physics Research Section A*, 629(Issue 1), February 2011.

[41] A. Valero et al. Optical buffer 1:16. Technical Report ATL-TILECAL-PUB-2007-005, CERN, Geneva, Mar 2007.

[42] S. Hass et al. Design and Performance of a PCI Interface with four 2 Gbit/s Serial Optical Links, 2004.

[43] M. Barczyk et al. Verification and Diagnostics Framework in ATLAS Trigger/DAQ. Technical Report ATL-DAQ-2003-033, CERN, Geneva, May 2003. DVS Web pages:http://atddoc.cern.ch/Atlas/DaqSoft/components/diagnostics/Welcome.html.

[44] W. E. Cleland and E. G. Stern. Signal processing considerations for liquid ionization calorimeters in a high rate environment. *Nuclear Instruments and Methods in Physics Research A*, 338:467–497, Jan 1994.

[45] M. C. M. Fiolhais. Correlated noise unfolding on a hadronic calorimeter. Technical Report ATL-TILECAL-PROC-2011-003, CERN, Geneva, Feb 2011.

[46] E. Fullana et al. Digital Signal Reconstruction in the ATLAS Hadronic Tile Calorimeter. *Nuclear Science, IEEE Transactions on*, 53(4):2139–2143, August 2006.

[47] A. Valassi et al. COOL, LCG Conditions Database for the LHC Experiments: Development and Deployment Status. LHC: Large Hadron Collider. Technical Report CERN-IT-Note-2008-019, CERN, Geneva, November 2008.

[48] C. Clement and P. Klimek. Identification of Pile-up Using the Quality Factor of Pulse Shapes in the ATLAS Tile Calorimeter. Technical Report ATL-TILECAL-PROC-2011-014, CERN, Geneva, Nov 2011.

[49] G. P. Salam M. Cacciari and G. Soyez. The anti- k t jet clustering algorithm. *Journal of High Energy Physics*, 2008(04):063, 2008.

List of Figures

List of Tables